

T.C.

ÇANAKKALE ONSEKİZ MART ÜNİVERSİTESİ

FEN BİLİMLERİ ENSTİTÜSÜ

YÜKSEK LİSANS TEZİ

BULANIK VERİTABANLARI VE SQL' DE

BULANIK MANTIK UYGULAMALARI

Vügar SALAHLI

BİLGİSAYAR MÜHENDİSLİĞİ ANABİLİMDALI

Tezin Sunulduğu Tarih: 08.02.2010

Tez Danışmanı:

Doç. Dr. Mehmet Ali SALAHLI

ÇANAKKALE

İÇERİK

	Sayfa
TEZ SINAVI SONUÇ BELGESİ.....	ii
İNTİHAL (AŞIRMA) BEYAN SAYFASI.....	iii
TEŞEKKÜR.....	iv
SİMGELER VE KISALTMALAR.....	v
ÖZET.....	vii
ABSTRACT.....	ix
BÖLÜM 1- GİRİŞ.....	1
BÖLÜM 2- BULANIK KÜMELER TEORİSİ VE BULANIK MANTIK.....	2
2.1. Bulanık kümeler teorisinin ve bulanık mantığın tarihçesi. Temel kavramları.....	4
2.1.1. Klasik kümeler.....	5
2.1.2. Bulanık kümeler.....	7
2.1.3. Normal küme.....	9
2.2. Bulanık değerlerin grafiksel ifade biçimleri.....	10
2.3. Bulanık küme işlemleri.....	11
2.3.1. Bulanık birleşim.....	13
2.3.2. Bulanık kesişim.....	13
2.4. Bulanık kümeler üzerinde mantık işlemleri.....	15
2.4.1. Bulanık VE.....	15
2.4.2. Bulanık VEYA.....	15
2.4.3. Bulanık DEĞİL.....	16
BÖLÜM 3- BULANIK İLİŞKİLER CEBİR.....	16
3.1 Bulanık ilişkisel cebir işlemleri.....	16
3.1.1. Birleşim ($r \cup s$).....	16
3.1.2. Kesişim ($r \cap s$).....	18
3.1.3. Fark ($r - s$).....	19

3.1.4. Kartezyen çarpım (r X s)	19
3.1.5. İzdüşümü (P _x (r))	20
3.1.6. Bitiştirme	20
3.2. Benzerlik ilişkileri.....	21
3.2.1. İzdüşümü	27
3.2.2. Seçme	27
3.1.6. Doğal bitiştirme.....	28
BÖLÜM 4- BULANIK VERİ TABANLARI.....	30
4.1. Bulanık Veritabanlarının geliştirilmesi yöntemleri	30
4.2. Bulanık Veritabanı uygulamaları	33
4.3. Bulanık İlişkisel Veritabanlarının teori temeli.....	36
4.4. Bulanık Veritabanlarının tasviri ve modelleştirilmesi.....	40
4.5. Bulanık sorguların ve verilerin işlenilmesi	41
4.6. Bulanık Veritabanlarının çalıştırılması.....	43
4.7. Bulanık Veritabanlarında sorgu dili.....	45
4.7. Bulanık Varlık İlişki Modeli.....	47
BÖLÜM 5- BULANIK SQL UYGULAMASI.....	50
5.1. Ana Ekran	51
5.2. Veritabanı oluşturma sihirbazı	51
5.3. Bulanık mantık işlemler programı	54
5.4. Bulanık Sorgulama	55
5.5. Klasik ve bulanık sorgulamaların karşılaştırılması	55
BÖLÜM 6- SONUÇ	58
KAYNAKLAR.....	60
Çizelgeler.....	I
Şekiller	III
Özgeçmiş	V

ÖZET

**BULANIK VERİTABANLARI VE SQL' DE BULANIK MANTIK
UYGULAMALARI**

Vügar SALAHLI

Canakkale Onsekiz Mart Üniversitesi

Fen Bilimleri Enstitüsü

Bilgisayar Mühendisliği Anabilim Dalı Yüksek Lisans Tezi

Danışman: Doç. Dr. Mehmet Ali SALAHLI

08.02.2010, 61

Veritabanları, bilgisayar sistemlerinin önemli bileşenlerindedir. Klasik veritabanı sistemleri yalnızca iyi tanımlanmış ve kesin veriler üzerinde işlem yapmaya izin verir. Fakat gerçek yaşamda pek çok uygulama bulanık, iyi tanımlanmamış, kesin olmayan verilerin de işlenilebilmesini gerektiriyor. Bu tür uygulamalara bilgisayarlı tasarım, çoklu ortam, coğrafi bilgi sistemleri vs. alanlardaki uygulamaları örnek göstermek mümkündür. Bu uygulamalarda karmaşık nesnelere ve onların arasındaki anlamsal ilişkilerin modellenmesi, bulanık bilgilerin işlenmesine bağlıdır.

Bulanık mantık, 1980 'li yıllardan itibaren veri modellerinin geliştirilmesi için kullanılmaya başladı. Bulanık mantığın veri modellemesine ilave edilmesinde amaç; belirsiz ve kesin olmayan bilgileri veri modellerinde ifade edebilme ve işleyebilme imkânının oluşturulması idi.

Bu tez çalışmasında bulanık veritabanlarının bazı teori ve pratik yönleri açıklanmıştır. Bulanık veritabanı işlemlerini anlatmak için yazılım uygulaması geliştirilmiştir.

Bulanık veritabanları araştırmalarının çağdaş durumu öğrenilmiştir ve bulanık küme işlemlerinin ayrıntılı açıklaması verilmiştir.

Tez çalışmasında klasik ve bulanık ilişkisel veri modelleri kavramları ve tanımları anlatılmıştır. Bulanık birleşim, bulanık kesişim, bulanık seçme, bulanık izdüşümü, benzerlik ilişkileri gibi işlemler anlatılmıştır. Bu işlemler klasik cebir işlemleri ile karşılaştırılmıştır.

Tezde bulanık mantıksal veritabanlarına da yer verilmiştir. Kısaca bulanık varlık ilişki modeli ve bulanık genişlenmiş varlık-ilişki modeline değinilmiştir.

Bulanık verileri yönetmek ve işlemek için SQL sorgulama dili üzerinde geliştirilmiş bulanık SQL dillerinin yapısı ve işlevleri kısaca anlatılmıştır.

Anahtar sözcükler: Bulanık mantık, Bulanık SQL, Bulanık arama.

ABSTRACT

FUZZY DATABASES AND FUZZY LOGIC APPLICATIONS IN SQL

Vugar SALAHLI

Canakkale Onsekiz Mart University

Graduate School of Science and Engineering

Chair for Computer Engineering Thesis of Master of Science

Advisor: Doc.Dr. Mehmet Ali SALAHLI

08.02.2010, 61

Databases are a very important component in computer systems. Classical database systems can only deal with well-defined and unambiguous data. In the real world, however, there exist uncertain or imprecise data, which cannot be defined in certain and well-defined form by any means. In everyday life we often make decisions based on a database which can represent and manipulate fuzzy data. Most of databases applications in CAD/CAM, multimedia, geographic information systems, and etc. require the modeling and manipulation of complex objects and semantic relationships with imprecision and uncertainty.

Since the early 1980's fuzzy logic has been used to extend various data models. The purpose of introducing fuzzy logic in database modeling is to enhance the classical models such that uncertain and imprecise information can be represented and manipulated.

In the thesis theoretical and practical aspects of fuzzy databases are given. The software application for demonstration some fuzzy database operations was developed.

Current state of the fuzzy database researches is learned. Detailed description of the fuzzy set operations is given.

In the thesis the basic definitions and concepts related to the classical and fuzzy relational models are explained. Fuzzy relational operations, such as union, intersection, projection, selection, similarity relations and etc. are described. Comparisons of this operations with classic operations is given.

The logical fuzzy database models is discussed in the thesis. Fuzzy entity – relationship model and the enhanced entity-relationship model are introduced briefly.

A structures and functions of the fuzzy extensions of the SQL language, which allow to handle fuzzy data are explained in the thesis briefly.

Keywords: Fuzzy logic, Fuzzy SQL, Fuzzy search.

BÖLÜM 1

GİRİŞ

Bulanık kümeler teorisi belirsizlik durumunda modelleme ve sonuç alma için çok etkili bir yöntemdir. Bulanık kümeler teorisi ve bulanık mantık kavramı ilk kez Berkeley Üniversitesinin (Kaliforniya, ABD) profesörü Lütfi Asker Zade (Lotfi Asker Zadeh) tarafından 1965 yılında önerilmiştir (Zadeh, 1965). Bulanık kümenin elementleri, yalnız 0 ve 1 değerleri alabilen klasik küme elementlerinden farklı olarak 0 ve 1 arasında herhangi bir değer alabilir. Klasik mantık, klasik kümeler teorisine; bulanık mantık ise bulanık küme teorisine dayanmaktadır.

Bulanık mantığın ilk sanayi uygulaması bulanık denetleyiciler alanında olmuştur. 1980 yıllarında Danimarkalı mühendisler tarafından çimento sobalarının bulanık denetleyicisi geliştirilmiştir; ama bu uygulama Batıda yankı uyandırmamıştır. Bulanık mantık Batının aksine Japonya’da daha çok ilgi görmüştür. Japonlar yeraltı trenlerinin hareketini otomatik denetleyen bulanık sistem geliştirmiş ve uygulamışlardır. Başarılı sonuçlar bulanık mantığın diğer alanlarda da uygulanmasına sağlamıştır. Asansör kontrolü sistemleri ve havalandırma sistemleri buna örnek olabilir. 1990’larda Japonlar çamaşır makinesi, elektrik süpürgesi ve araba gibi tüketici ürünlerinde de bulanık mantığa dayalı denetim sistemlerini uygulamışlardır. Bu uygulamalar sonunda Avrupa ve Birleşik Devletlerde de bulanık mantık yöntemleri dikkat çekmiştir.

Bulanık mantık sistemleri uygulamaları 1990’ların başından itibaren yeni bir ivme kazanmıştır. Özellikle bilgi sistemlerinde, sıradan veritabanları ve bilgi arama alanlarında gelişmeler yaşanmıştır. ABD ’de Makine Zekâsı Enstitüsünde Profesör R.R.Yager ’in danışmanlığında bulanık mantığa dayalı ilk arama motoru geliştirilmiştir (Larsen, 2005).

İnternet ve web, bulanık mantık uygulamalarına ilgiyi artırmıştır. İnternet tabanlı toplumda kararların alınmasında elektronik ortamı bilgilere kesin ihtiyaç duyulmaktadır; ama bu bilgiler çoğunlukla tam ve belirgin biçimde sunulmaz ve onların kesin güvenilirliği bulunmaz. Öte yandan bu bilgilerden yararlanmak için, onların kullanıcı tarafından işlenebilir olması gerekir. Ortaya çıkan bilgi yoğunluğu karşısında bu bilgilerin bulunması, değerlendirilmesi, kullanıcının isteklerini en iyi şekilde karşılayacak bilgilerin

çıkarılması ve karar oluşturma sistemlerinde kullanılması için etkili bir bilgi işleme ve depolama aracına ihtiyaç duyulmaktadır. Bu bağlamda, belirsiz bilgileri sunma ve işleme imkânı sağlayan bulanık mantık bilgi keşfi, veri madenciliği uygulamalarında bulanık mantık etkili biçimde kullanılmaktadır.

Bugün matematik, iktisat, sanayi ve gen mühendisliği, uzay bilimciliği ve birçok farklı alanlarda bulanık mantık uygulamaları geniş yer bulmuştur. Görüntü ve ses tanıma, uzman sistemleri geliştirme, karar oluşturma, kontrol, sınıflandırma, öğrenme gibi sorunların çözümünde bulanık mantık yaygın olarak kullanılmaktadır. Son yıllarda ticari ve endüstriyel alanlarda binlerce bulanık sistem uygulamaları gerçekleştirilmiştir.

Bilgi Sistemlerinde bulanık kümeler teorisinin uygulanması ile bağlı bir yön de Bulanık veri tabanlarının geliştirilmesi ile bağlıdır.

Hiyerarşik modeller ve ağ modelleri üzerinde geliştirilmeye başlamış veritabanları sistemleri, ilişkisel veri modeli ile yeni bir gelişme evrenine geçmiştir. Daha sonraları nesneye yönelik ve nesne ilişkisel modelleri geliştirilmiştir. Bu modeller, çoklu ortam verileri gibi çok karmaşık veri yapılarını da ifade etme ve işleme imkânı sağlar; ama gerçek dünya uygulamaları çok daha karmaşık verilerin tasvirini ve işlenilmesini gerektirir. Klasik veri modelleri ise bunları gerçekleştirmekte yetersiz kalır. Özellikle; kesin olmayan, eksik olan bilgilerin işlenilmesi çözülmesi gereken bir sorun olarak ortaya çıkmaktadır. Bu sorunun çözümü için en etkili ve yaygın yaklaşımlardan birisi bulanık veri tabanlarının geliştirilmesidir.

İlk bulanık veritabanı uygulamalarına geçen asrın 70. yıllarının sonlarında rastlanmıştır. Geçen 40 yıl içinde bulanık veri tabanları ve bulanık bilgi işlemlerinde önemli gelişmeler sağlanmıştır. Tüm veritabanı modelleri; ağ, ilişkisel, varlık –ilişkisel ve nesneye yönelik modeller bulanık kümeleri kullanmakla bulanık işlemleri gerçekleştirebilme yönünde genişletilmiştir. Günümüzde, coğrafi bilgi sistemleri ve anlamsal web alanlarında bulanık veritabanı uygulamaları ilgi çekmektedir.

Bulanık küme teorisinin veri tabanlarında, bilgi depolama ve bilgi arama sistemlerinde kullanılmasının gerekliliği kesin olmayan bilgilerin işlenilmesi ile bağlıdır. Kesin olmayan bilgiyi işleyebilen veritabanı aynı zamanda öznel uzman fikirlerini, muhakemelerini ve sözel terimlerle ifade edilmiş değerleri de işleyebilir. Bu tür bilgiler, özellikle tıbbi teşhis koyma, jeoloji arama, yatırım, belge arama gibi alanlarda çok önemlidir. Pek çok uygulamalarda kesin, muğlâk, kesin olmayan veriler üzerinde arama yapılması daha etkili sonuçlar vermektedir. Örneğin; “çalışanlar içerisinde daha yüksek

eğitilmiş ve yeterli derecede deneyimli olanların bulunması”, şehir merkezine yakın ve fiyatı makul olan otellerin bulunması gibi” sorguların cevapları kesin veriler içeren sorgu cevaplarına nispeten daha kapsamlı olacaktır.

Kaynaklardaki; bulanık ilişkisel veri tabanları (BİVT) ile bağlı araştırmalar, bulanık veri tabanlarının modellenmesi ve ifade edilmesi, anlamsal ölçmeler, veri bağımsızlığı, sorguların ve verilerin işlenmesi, veri tabanlarının çalıştırılması gibi meseleleri kapsamaktadır.

Bu tez çalışmasının amacı;

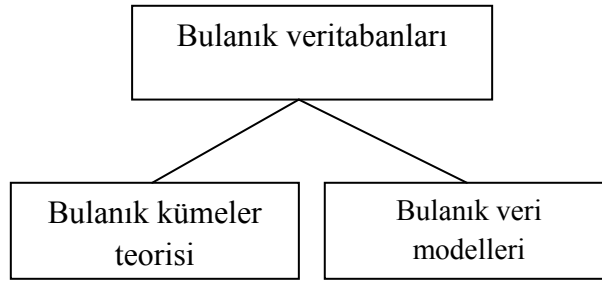
- Bulanık veri tabanlarının geliştirilmesi ile ilgili sorunları incelemek,
- Bulanık veri tabanlarında verilerin ifade edilmesi ve işlenmesi, veri tabanlarında bulanık sorguların işlenmesi, bulanık ilişkisel işlemlerle bağlı sorunları öğrenmek,
- Bulanık veri tabanlarının oluşturulması ve bu veritabanında SQL (Structured Query Language) sorgulama dili esasında bulanık verileri ifade edebilen ve temel sorgu işlemlerini gerçekleştirebilen sistem geliştirmektir.

Tezin amacına uygun olarak yapılan çalışmalar tezin aşağıdaki bölümlerinde açıklanmıştır. Giriş bölümünde tezin amacı, yapılan çalışmalar ve tezin yapısı hakkında bilgi verilmiştir. Tezin ikinci bölümünde bulanık kümeler teorisi ve bulanık mantık kavramları incelenmiş, bulanık küme işlemleri öğrenilmiş, bulanık değerlerin ifade biçimleri verilmiştir. Tezin üçüncü bölümünde bulanık ilişkisel cebir işlemleri ayrıntılı olarak incelenmiştir. Tezin dördüncü bölümü bulanık veri tabanlarının geliştirilmesi ile ilgilidir. Bu bölümde önce bulanık veri tabanlarının geliştirilme mantığı, teori tabanı incelenmiştir. Bulanık veri tabanlarının kavramsal tasarımı, özellikle genişletilmiş varlık-ilişki modeli üzerinde bulanık ilişki ve özelliklerin ifade edilmesi anlatılmıştır. Daha sonra Bulanık verilerin veri tabanlarında ifade edilmesi yöntemleri araştırılmıştır. İlişkisel veri modelleri üzerinde bulanık sorguların gerçekleştirilmesi yollarına bakılmıştır. Geliştirmiş olduğum bulanık veritabanı uygulaması son bölümde anlatılmıştır. Uygulama yazılımının mantıksal yapısı verilmiş, işleyişi anlatılmıştır. Bulanık sorgular esasında sistemin çalışması örnekleri verilmiştir.

BÖLÜM 2

BULANIK KÜMELER TEORİSİ VE BULANIK MANTIK

Bulanık veritabanlarının temelini iki bileşen oluşturmaktadır. Bunlar bulanık küme teorisi ve veritabanı modelleridir (Şekil 2.1). Bu bölümde; bulanık kümeler teorisinin temel kavramları anlatılacak, bulanık kümelerle klasik kümelerin karşılaşması yapılacak, bulanık kümeler üzerinde yapılan küme ve mantık işlemleri öğretilenektir.



Şekil 2.1. Bulanık Veritabanının temel bileşenleri.

2.1. Bulanık Kümeler Teorisinin ve Bulanık Mantığın Tarihçesi, Temel

Kavramları

1965 yılında önermiş olduğu bulanık kümeler teorisi ile aslen Azeri olan Berkeley Üniversitesinin profesörü Lotfi Zadeh (Lütfi Askerzade) dünya mantık, mühendislik ve bilim tarihine kendi adını yazdırdı. Zadeh, savaş gemilerindeki elektronik aygıtlarla yakınlaşan uçakları tanımlamaya imkân sağlayacak yazılım sistemleriyle ilgilenirdi. Uçaklar, tanınmalarını engelleyecek araçlarla donatılmıştır. Buna rağmen onları güvenli bir biçimde tanıyacak sistemlere ihtiyaç vardır. Bu durumlarda alınan bilgiler kesin değil, bulanıktır. Bu bilgiler, mantık dili ile söylersek kesin doğru veya yanlış değildir. Biraz doğru veya biraz yanlış olarak değerlendirilebilen bilgilerdir. Böylece, Zadeh bu tür kesin olmayan bilgileri işlemek için klasik mantığın genişlemesi olarak bulanık mantığı önerdi (Petry, 1996). Günlük hayatımız gerçekten belirsiz, kesin olmayan bilgilerle doludur. Biz yeni arabalar, *yüksek* binalar hakkında konuşuyoruz. Havanın *çok sıcak* olduğunu ve sınavların *zor* olduğunu söylüyoruz. Bu konuşmalarımızda geçen “*yeni*”, “*yüksek*”, “*çok sıcak*”, ve “*zor*” kavramları kesin olmayan niteleyicilerdir. Bulanık mantık, insana kesin olmayan niteleyicilerle muhakeme yürütme, karar verme sürecinin bilgisayarda

modelleştirilmesi imkânını sağlıyor. Bulanık mantıkta kesin sonuç almaya, yaklaşık sonuç almanın sınırlandırılmış hali gibi bakılıyor. Bu mantıkta her şey belirli bir derece ile doğrudur. Geleneksel mantıkta ve programlamada değerler kesindir. Eğer bize değişkenin kesin değeri belli değilse sonuç alabilme ve hesaplama yapabilme imkânımız yoktur. Bulanık mantığı ise bulanık kümelerle işliyoruz. Burada her bir değişkenin, bulanık kümeye belirli bir üyelik derecesi ile ait olan değeri vardır. 0 derecesi üyeliğin olmadığını, 1 derecesi ise üyeliğin tam olduğunu gösteriyor. Bu dereceler arasındaki her bir sayı belirsizliğin derecesini ifade eder. Bulanık küme, bakılan soruna bağlı olarak farklı biçimlerde sunulabilir.

Bir örneğe bakalım.

$$\text{Yaşlı (v)} = \begin{cases} 0, \text{ eger } v < 20 \\ (v - 20) / 40, \text{ eger } 20 \leq v \leq 60 \\ 1, \text{ eger } v > 60 \end{cases}$$

Bu örnekten şunu anlıyoruz ki; yaşı 20’den küçük olanlar yaşlı değil; ancak yaşı 60’dan yüksek olanlar kesinlikle yaşlıdır. Diğer kişiler için ise “yaşlı” olma derecesi gösterilen formül ile hesaplanmaktadır. Örneğin, 45 yaşlı bir insanın “yaşlı” olma derecesi 0.625’dir (Petry, 1996).

Bulanık kümeler teorisi ile belirgin olmayan kavram ve ilişkileri kesinlikle sunabiliriz. Örneğin; bulanık mantık her hangi bir kitabın, makalenin hangi derece ile verilmiş konuya ait olduğunu söyler. Bulanık kümeler teorisi ve bulanık mantık, yalnız belirgin olmayan bilgileri ifade etmekle yetinmiyor, aynı zamanda bu bilgileri işleyebiliyor. Bu, bulanık kümeler teorisi ve bulanık mantığın klasik mantığın yerine geçtiği anlamını vermez. Tam aksine, bulanık küme teorisi ve bulanık mantık klasik teorilere dayanır ve onların gelişimi ile yeni yöntemler kazanır. Klasik yöntemler ve bulanık mantık yöntemleri pek çok uygulamada (örneğin veri madenciliğinde) birlikte kullanılmaktadır. Bulanık küme teorisinin temel kavramlarını öğrenmeden önce klasik kümelere kısaca göz atalım.

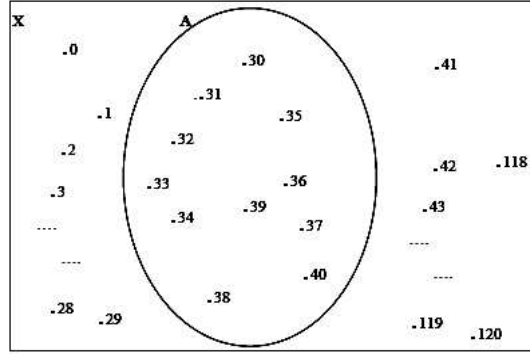
2.1.1. Klasik Kümeler

Klasik anlamda küme, nesnelerin iyi tanımlanmış listesidir. Kümeyi oluşturan öğelere, kümenin elemanı denir. a elemanı A kümesine ait ise $a \in A$ biçiminde yazılır ve “a, A kümesinin elemanıdır” gibi okunur. b elemanı A kümesine ait değilse, $b \notin A$

biçiminde yazılır ve “*b, A kümesinin elemanı değildir.*” gibi okunur. Varsayalım ki, x elemanları 0 ile 120 arasındaki yaş gurubunu temsil eden X evrensel kümesini oluşturur: $X=[0,120]$. A kümesi, bu X evreni üzerinde bir alt küme olsun ve 30 ile 40 yaş arasındaki yaşları temsil etsin: $A=[30,40]$. A kümesinin liste gösterimi

$$A=\{30, 31, 32, 33, 34, 35, 36, 37, 38, 39, 40\}$$

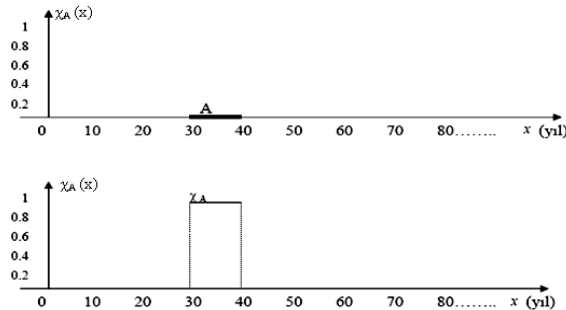
gibi, şematik gösterimi de şekil 2.2. de gösterildiği gibi olacaktır.



Şekil 2.2. X ve A kümelerinin ifadesi.

Listede ve şekilde görüldüğü gibi, örneğin 34 sayısı A Kümesinin elemanıdır, $34 \in A$. 43 sayısı ise A kümesinin elemanı değildir, $43 \notin A$.

Klasik kümeler teorisine göre bir sayı bir kümenin ya elemanıdır, ya da değildir. Başka bir deyişle bir nesne kümeye tam üye ise üyelik derecesi 1’dir veya üye değilse üyelik derecesi 0’dır. Bu yüzden klasik kümelerde elemanların üyelik dereceleri değer kümesinde $\{0,1\}$ şeklinde iki değer alabilir. Yaş grubunu temsil eden küme örneğini yeniden ele alalım. $X=[0,120]$, $A=[30,40]$. Klasik kümelerde bir x yaş değerinin A yaş gurubu kümesine üye olup olmadığını gösteren bağıntı $\chi_A(x)$ karakteristik fonksiyonu ile verilir.



Şekil 2.3. A kümesi ve x değerlerinin bu kümeye ait olup olmadığını gösteren $\chi_A(x)$ ilişki fonksiyonu

Evrensel kümeyi tanım kümesi, üyelik değerlerini ise değer kümesi olarak kabul edersek, bir $\chi_A(x)$ karakteristik fonksiyonu, $X=[0,120]$ tanım kümesinin, iki elemanı olan $\{0,1\}$ değer kümesine yansımısını ifade eder.

Herhangi bir x değerinin X evreni üzerindeki A klasik altkümeye üyelik ifadesi $\chi_A(X)$ matematiksel olarak aşağıdaki biçimde bir üyelik fonksiyonu ile gösterilir:

$$\chi_A(x) = \begin{cases} 1, & x \in A \\ 0, & x \notin A \end{cases}$$

Bu denkleme göre A kümesinin üyelik ifadesi; $x=[30,40]$ değerleri arasında $\chi_A(x)=1$ değerini, bunun dışında ise $\chi_A(x)=0$ değerini alan bir fonksiyondur. Evrensel küme üzerindeki bir x değerinin, bu kümenin bir alt kümesi olan A kümesine aitlik derecesini ifade eden bu ilişki fonksiyonuna “üyelik fonksiyonu” denir. Klasik kümeler teorisinde bir değer kümeye üyelik derecesi 1, yani tam (%100) veya 0, yani yok (%0)’tur.

2.1.2. Bulanık Kümeler

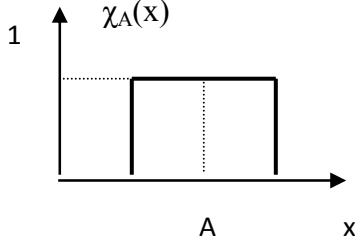
Gerçek dünyanın karmaşıklığı; incelenen kavramların kesin bir noktadan ziyade bir bölgeyi tanımlaması, ölçme sonuçlarının tek değil, aynı anda birden fazla değere sahip olması, yorumlanan olayların sanıldığı gibi yalın bir anlam değil, birbiri içine geçmiş birden fazla anlamı barındırmasından kaynaklanır. Bu duruma belirsizlik denir.

Bir olguyu ifade ederken, dilimizin doğasından gelen bir belirsizliği hemen fark edebiliriz. “*Zeynep akıllıdır*” veya “*Serkan gençtir*” gibi ifadeleri buna örnek olarak verebiliriz. Bunun da ötesinde kesin olarak ölçebildiğimizi sandığımız pek çok değer aslında ne kadar kesindir? Bir voltmetrenin ölçtüğü gerilim 5V olsun. Bu gerilim gerçekten 5V mudur? Yoksa 4,999V veya 5,019V olmasına karşın çözümlüğe bağlı olarak bahsettiğimiz değere mi yuvarlanmaktadır?

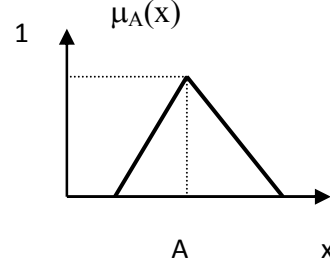
Bulanık kümeler, bu tür belirsiz ifadeleri modelleyen matematiksel işlevlerdir. Bulanık küme teorisi; eldeki bilgilerin belirsiz, eksik, bulanık olduğu ve kesin olmadığı durumlarda bu bilgileri karşılaştırmak ve değerlendirmek üzere matematiksel araçlar sağlar. *Sıcak, soğuk, hızlı, yavaş, az, çok* gibi sözel değişkenler bu sayede bilgisayarların hesap yapabileceği bulanık kümelerdir.

Zadeh, klasik kümeler teorisinde sadece iki farklı üyelik derecesi alan ifadeyi 0 ile 1 arasında çeşitli üyelik dereceleri alabilen bir başka gösterim şekliyle genişletti ve bu yeni üyelik fonksiyonunu $\mu(x)$ ile ifade etti ($\mu(x): X \rightarrow [0,1]$). Bu nedenle klasik kümeleri

bulanık kümelerin özel bir durumu olarak kabul edebiliriz. İki küme türünde de kümelerin sınırları dışında üyelik dereceleri sıfırdır; çünkü kümelerin buralarda elemanları yoktur.



Şekil 2.4.a. Kesin değerli A kümesi



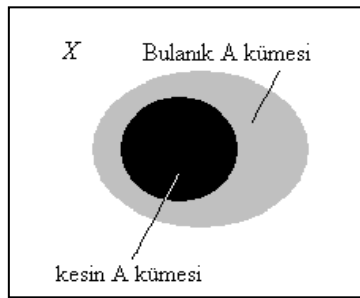
Şekil 2.4.b. Bulanık değerli A kümesi

Fakat kesin değerli bir A kümesinden farklı olarak bir A bulanık kümesinin elemanları $[0,1]$ aralığında 0 veya 1'den farklı (0.2,0.5,0.7...) gibi dereceleri de içine alan üyelik değeri alabiliyorlar:

$$\mu_A(x) \in [0,1]$$

$\mu(x)$ üyelik fonksiyonunun x 'e bağlı olarak aldığı sayısal değere bundan sonra üyelik derecesi diyeceğiz.

Klasik kümeleri tek bir parametre ile, yani elemanları ile ifade etmek yeterlidir; çünkü bir x değeri kümenin elemanıysa ($x \in A$) üyelik derecesinin 1 olduğunu belirtmemize gerek yoktur. Bir bulanık küme ise, klasik kümeden farklı olarak iki parametre ile ifade edilebilir; çünkü küme, elemanlardan ($x_i \in A$) ve bu elemanların o kümeyle ne kadar ait olduğunu gösteren üyelik derecelerinden $\mu_A(x_i)$ oluşur.



Şekil 2.5. Kesin ve bulanık A kümeleri

Varsayalım ki, X boş olmayan kümedir. X 'e dâhil olan A bulanık kümesi, bu kümenin üyelik fonksiyonu ile ifade olunur;

$$\mu_A: X \rightarrow [0,1]$$

$\mu_A(x)$ - her bir $x \in X$ için x elementinin A bulanık kümesinin üyelik derecesini ifade eder. A kümesi bütünlükle

$$A = \{ (u, \mu_A(u)) \mid u \in X \}$$

çifti ile ifade edilebilir.

Sözü geçen X evreni $X = \{x_1, x_2\}$ sonlu küme, onun üyeleri ve üyelik dereceleri ayrık ise A kümesi X evreninde genel olarak aşağıdaki gibi ifade edilir:

$$A = \mu_1/x_1 + \dots + \mu_n/x_n$$

Burada μ_i/x_i , $i = 1, \dots, n$ ifadesi μ_i 'nin, X_i 'nin üyelik derecesi olduğunu gösteriyor, "+" işareti ise birleşimi ifade ediyor.

X evreni sürekli ise A kümesinin ifadesi böyle oluyor:

$$\tilde{A} = \{ \int \mu_A(x)/x \}$$

X evrensel konuşma alanındaki tüm bulanık kümeler $F(X)$ ile gösterilir (<http://users.abo.fi/rfuller/nfs1.pdf>).

2.1.3. Normal küme

$A(x)=1$ eşitliğini sağlayan öyle bir $x \in X$ varsa klasik X kümesinin A bulanık altkümesine normal küme denir. Aksi halde A normal küme değildir.

Bulanık kümelerin en önemli kavramlarına destek ve α -kesim kavramlarını da aittir.

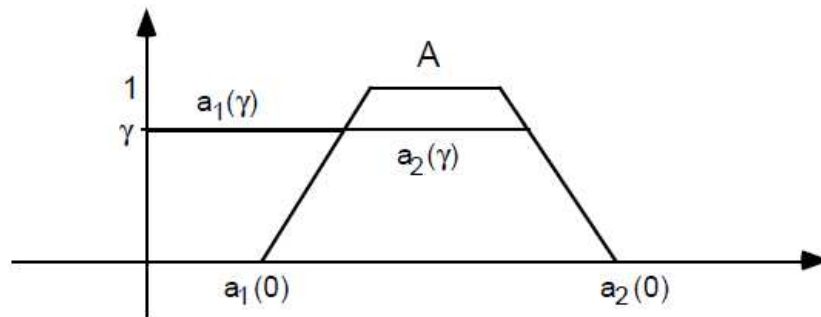
Destek. Varsayalım ki; A , X evreninde bulanık altkümedir. A kümesinin desteği $\text{supp}(A)$ ile gösterilir ve X' in, A' ya üyelik dereceleri sıfır olmayan tüm elementlerinden oluşan kesin altkümedir (<http://users.abo.fi/rfuller/nfs1.pdf>):

$$\text{Supp}(A) = \{ x \mid x \in X \text{ ve } \mu_A(x) > 0 \}$$

O zaman bulanık A kümesini aşağıdaki gibi yazılabilir:

$$A = \{ \mu_A(x)/x \mid x \in \text{Supp}(A) \}$$

Bu ifadenin anlamı: "Yalnız üyelik fonksiyonları sıfırdan büyük olan bulanık üyeler A kümesine dâhildirler." şeklindedir.



Şekil 2.6. A bulanık kümesinin desteği $[a_1(0), a_2(0)]$

Örnek: Fiyatın belirlenmesi için “YÜKSEK” terimini bulanık küme gibi ifade edelim:

$$YÜKSEK = [0.0/1, 0.125/5, 0.5/10, 0.8/25, 0.9/50, 1.0/100]$$

Bu kümede 0.0/1 üyesi destek kümesinin üyesi değildir:

$$\text{Supp}(A) = [0.125/5, 0.5/10, 0.8/25, 0.9/50, 1.0/100]$$

Destekle sıkı bağlı bir kavram α -kesim kavramıdır.

α -kesim. X'in A bulanık kümesinin α -kesimi A_α ile gösterilen bulanık olmayan kümedir ve bu kümenin üyelerinin üyelik fonksiyonları, verilmiş α değerine eşit veya bu değerden büyüktür:

$$A_\alpha = \{t \in X \mid [A] \geq \alpha, \text{ eğer } \alpha > 0\}$$

Örnek. Varsayalım ki, $X = \{-2, -1, 0, 1, 2, 3, 4\}$ ve $A = \{0.0/-2 + 0.3/-1 + 0.6/0 + 1.0/1 + 0.6/2 + 0.3/3 + 0.0/4\}$

O zaman

$$A_{0.3} = \{-1, 0, 1, 2, 3\}$$

$$A_{0.3 < \alpha \leq 0.6} = \{0, 1, 2\}$$

$$A_{0.6 < \alpha \leq 1} = \{1\} \quad (\text{http://users.abo.fi/rfuller/nfs1.pdf}).$$

Her hangi kümenin α -kesimleri, uygun destek kümesinin altkümeleridir. α 'nın seçilmesi isteğe bağlıdır; ama çoğu zaman α 'nın değeri uygulamalarda evrenin gereken altkümelerini seçmek için belirlenir.

2.2. Bulanık değerlerin grafiksel ifade biçimleri

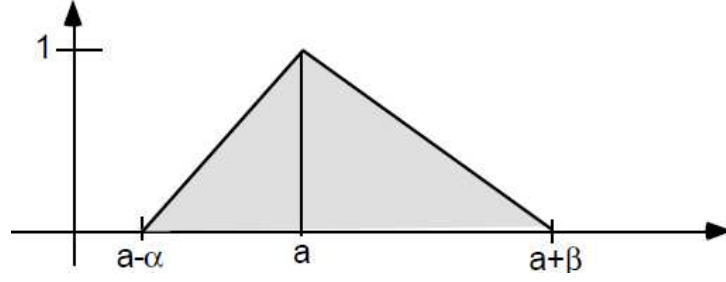
Bulanık kümelerin yorumlanmasında onların grafiksel ifade biçimlerini kullanmak etkili olabilir. Bulanık kümeleri sigmoid, gaussian, pi gibi tipik fonksiyonlarla ifade etmek mümkündür; ama bu fonksiyonlar bilgisayarda işlem zamanını yükseltir. Bu bakımdan çoğu uygulamalarda doğrusal fonksiyonlar tercih ediliyor.

A bulanık kümesinin üyelik fonksiyonu aşağıdaki formül ile ifade edilebilirse, ona üçgen bulanık sayı denir:

$$A(t) = \begin{cases} 1 - \frac{a-t}{\alpha} & \text{eğer } a - \alpha \leq t \leq a \\ 1 - \frac{t-a}{\beta} & \text{eğer } a \leq t \leq a + \beta \\ 0 & \text{eksi halde} \end{cases}$$

Burada a- merkez, $\alpha > 0$ sol kenar, $\beta > 0$ sağ kenardır.

Üçgen biçimindeki bulanık sayı için $A = \{a, \alpha, \beta\}$



Şekil 2.7. Bulanık kümenin üçgen ifadesi

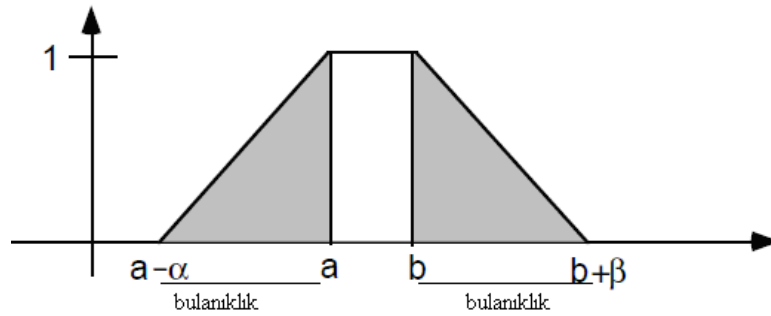
Merkezi a olan üçgen bulanık sayımı “x, yaklaşık olarak a 'ya eşittir” bulanık niceliği ile ifade etmek mümkündür.

A bulanık kümesinin üyelik fonksiyonu aşağıdaki ifade ile hesaplanırsa, ona dayanıklılık aralığı [a,b], sol kenarı α , sağ kenarı β olan yamuksal bulanık sayı denilir:

$$A(t) = \begin{cases} 1 - (a-t)/\alpha & \text{eğer } a - \alpha \leq t \leq a \\ 1 & \text{eğer } a \leq t \leq b \\ 1 - (t-b)/\beta & \text{eğer } a \leq t \leq b + \beta \\ 0 & \text{eksi halde} \end{cases}$$

O halde biz (a,b, α , β) yazılışını kullanırız.

A'nın desteği (a- α , b+ β) olacak.



Şekil 2.8. Bulanık kümenin yamuk ifadesi

Yamuksal bulanık sayı “x yaklaşık olarak [a,b] aralığındadır” bulanık niceliği anlamındadır.

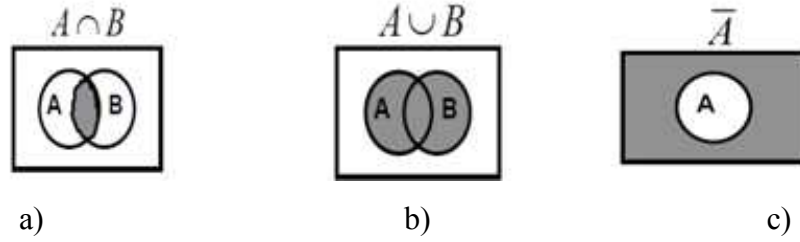
2.3.Bulanık küme işlemleri

Tüm temel klasik küme işlemlerinin bulanık kümeler için de karşılığı bulunmaktadır. Bunun dışında bulanık kümeler için üyelik derecelerine dayalı ve kesin kümelerde bulunmayan ilave işlemler de mevcuttur. Bulanık küme işlemlerinde kullanacağımız A ve B bulanık kümelerini ifade etmek için $\mu_A(x), \mu_B(x)$ üyelik fonksiyonlarını kullanacağız.

Kesin kümeler için temel işlemler bunlardır:

- Birleşim veya mantıksal *VEYA*,
- Kesişim veya mantıksal *VE*,
- Tamamlanma veya mantıksal *DEĞİL*

Şekil 2,9’da kesin küme işlemlerinin grafiksel ifadesi verilmiştir.



Şekil 2.9. Kesin kümelerde işlemlerin grafiksel ifadesi:

a) kesişim, b) birleşim, c) tamamlama

Benzer olarak, bulanık kümeler üzerinde işlem yapmaya olanak sağlayan bulanık tamamlama, kesişim ve birleşim işlemleri tanımlanmıştır; ama klasik küme ve bulanık küme teorisi işlemleri arasında farklar da mevcuttur.

Bulanık işlemler şöyle formüle edilebilir:

Eşitlik işlemi:

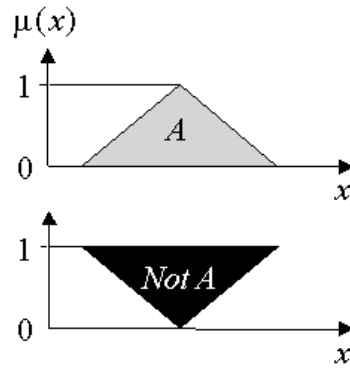
$$A = B \Leftrightarrow \mu_A(x) = \mu_B(x)$$

Kapsama işlemi:

$$A \subseteq B \Leftrightarrow \mu_A(x) \leq \mu_B(x)$$

Tamamlama işlemi:

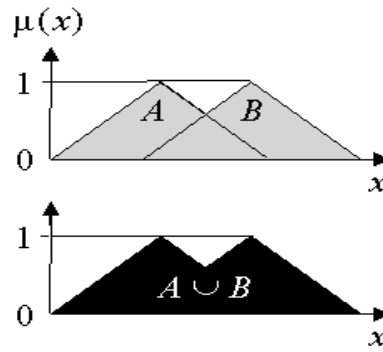
$$\sim A(x) = \{x / (1 - \mu_A(x))\}$$



Şekil 2. 10. Bulanık tamamlamanın grafiksel ifadesi

2.3.1. Bulanık Birleşim

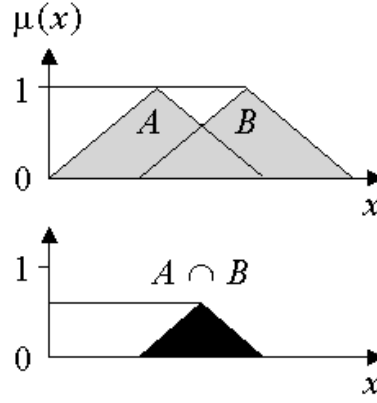
$$A \cup B \Leftrightarrow \mu_A(x) \cup \mu_B(x) = \max(\mu_A(x), \mu_B(x))$$



Şekil 2.11. Bulanık birleşim işleminin grafiksel ifadesi

2.3.2. Bulanık Kesişim

$$A \cap B \Leftrightarrow \mu_A(x) \cap \mu_B(x) = \min(\mu_A(x), \mu_B(x))$$



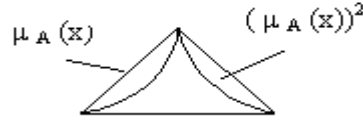
Şekil 2.12. Bulanık kesişim işleminin grafiksel ifadesi

Bunların dışında bulanık kümeler için yığılma (concentration) ve genişleme (dilation) işlemleri de kullanılmaktadır. Bu işlemler kesin kümeler için mevcut değildir. Yığılma ve genişleme işlemleri yaygın olarak sözel niceleyicileri ifade etmek için kullanılır.

Yığılma işlemi -CON(A): Bu işlem, üyelik dereceleri küçük olan elementler için üyelik derecelerini orantılı olarak daha da küçültmekle bulanık elementlerin yığılmasını gerçekleştirir.

$$\mu_{CON(A)}(x) = (\mu_A(x))^2$$

Yığılma işleminin grafiksel ifadesi şekil 2.13'de verilmiştir.

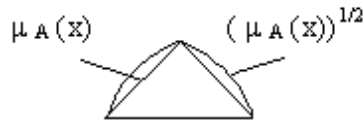


Şekil 2.13. Yığılma işleminin grafiksel ifadesi

Genişleme işlemi-DIL(A): Küçük üyelik dereceli elementlerin üyelik derecelerini orantılı olarak yükseltmekle bulanık elementlerin genişlemesini sağlar:

$$\mu_{DIL(A)}(x) = (\mu_A(x))^{1/2}$$

Genişleme işleminin grafiksel ifadesi şekil 2.14'de verilmiştir



Şekil 2.14. Genişleme işleminin grafiksel ifadesi

2.4. Bulanık kümeler üzerinde mantık işlemleri

Bulanık mantık işlemleri bulanık boolean değerlerinin tasvirine bağlıdır. Klasik mantıkta birleşim küme işlemi ile mantıksal VEYA işlemi arasında, kesişim işlemi ile VE arasında fark vardır; ama bulanık kümeler teorisinde mantık ve küme işlemleri arasında fark bulunmaz.

Bulanık birleşim \equiv Bulanık VEYA

Bulanık kesişim \equiv Bulanık VE

Bulanık tamamlanma \equiv Bulanık DEĞİL

Operantları, bulanık kümelerle ifade edilen üç mantık işlemi *bulanık değil*

($\bar{\cdot}$), *bulanık ve* ($\tilde{\cdot}$) ve *bulanık veya* (\vee) aşağıda tanımlanmıştır.

2.4.1. Bulanık VE

Bulanık ve işleminin sonucu bulanık boolean değerdir. *Bulanık ve* "kesişim" işlemi gibi tanımlanabilir.

$A: = \mu_A(x)$ ve $B: = \mu_B(x)$, U evreni üzerinde bulanık kümelerle ifade edilmiş iki boolean değer olsun. O zaman:

$$A \tilde{\cdot} B: \min(\mu_A(u), \mu_B(u)), u \in U.$$

2.4.2. Bulanık VEYA

Bulanık veya işleminin sonucu bulanık boolean değerdir. *Bulanık veya* "birleşim" işlemi gibi ifade edilebilir.

$A: = \mu_A(x)$ ve $B: = \mu_B(x)$, U evreni üzerinde bulanık kümelerle ifade edilmiş iki boolean değer olsun. O zaman:

$$A \vee B: \max(\mu_A(u), \mu_B(u)), u \in U.$$

2.4.3. Bulanık DEĞİL

Bulanık değil işleminin sonucu bulanık boolean değerdir. *Bulanık değil* işlemi kümeler üzerinde "tamamlama" işlemi gibi ifade edilebilir.

$\mu_A(u)$, U evreni üzerinde bulanık kümelerle ifade edilmiş boolean değerler olsun. O zaman A kümesi üzerinde *bulanık değil* işleminin sonucu şöyle olacak:

$$\bar{A}: (1 - \mu_A(u)), u \in U$$

BÖLÜM 3

BULANIK İLİŞKİLER ÜZERİNDE İŞLEMLER

Bu bölümde bulanık veritabanı modellerinde verilerin ve sorguların işlenilmesi için gereken bulanık cebir işlemlerini öğreneceğiz. İlişkisel veri modelleri üzerinde yapılan işlemleri iki sınıfa ayıracağız:

- İlişkisel cebir işlemleri,
- Benzerlik işlemleri.

3.1 Bulanık İlişkisel Cebir İşlemleri

Önce şunu belirtmek gerekiyor ki; klasik cebirde ilişkiler arasındaki eşdeğerlilik ve dönüştürme kuralları bulanık cebir için de geçerlidir. r, s R şeması üzere, u ise Q şeması üzere bulanık ilişkiler olsun. P_f, R' 'in özelliklerini kapsayan koşul yüklemidir. O halde;

$$a) r \cup s = s \cup r \text{ ve } r \cap s = s \cap r;$$

$$b) r \cup r = r \text{ ve } r \cap r = r;$$

$$c) r \cap (r \cup s) = r \text{ ve } r \cup (r \cap s) = r;$$

$$d) (r \cup s) \cup u = r \cup (s \cup u) \text{ ve } (r \cap s) \cap u = r \cap (s \cap u);$$

$$e) r \cap (s \cup u) = (r \cap s) \cup (r \cap u) \text{ ve } r \cup (s \cap u) = (r \cup s) \cap (r \cup u);$$

$$f) r \cup s = r \cup (s - r) \text{ ve } r \cap s = r - (r - s).$$

Bu kurallar bulanık birleştirme, birleşim ve kesişim işlemlerinde de uygulanabilir.

$$u \bowtie (r \cup s) = (u \bowtie r) \cup (u \bowtie s) \text{ ve } u \bowtie (r - s) = (u \bowtie r) - (u \bowtie s)$$

$$\sigma_{pf}(r \cup s) = \sigma_{pf}(r) \cup \sigma_{pf}(s) \text{ ve } \sigma_{pf}(r - s) = \sigma_{pf}(r) - \sigma_{pf}(s)$$

$$\sigma_{pf}(w \bowtie r) = w \bowtie \sigma_{pf}(r).$$

Klasik ilişkisel cebir işlemleri, bulanık kümeler üzerinde de yapılabilmektedir. Bu işlemler bulanık birleşim, bulanık kesişim, bulanık kartezyen çarpım, bulanık fark, bulanık birleştirme, bulanık bölme, bulanık izdüşümü ve bulanık seçme işlemleridir.

3.1.1. Birleşim ($r \cup s$)

Birleşim işlemi birleşebilir özelliği bulunan ilişkiler üzerinde gerçekleştirilebilir. Birleşebilirlik, işleme katılan ilişkilerin özelliklerinin ve özelliklerin sayısının aynı olmasını gerektiriyor. Klasik ilişkisel cebirde r ve s ilişkilerinin birleşimi olan $r \cup s$ öyle

bir ilişkidir ki, bu ilişkide r ve s’in tüm satırları yer alır. İlişkisel cebirin şartlarına uygun olarak satırlar aynı olursa, bu satırlardan yalnız birisi sonuç ilişkide bulunmuş olur.

R ve S, $X \times Y$ üzerinde tanımlanmış bulanık ilişkiler olsun.

R ve S bulanık ilişkilerinin birleşimi aşağıdaki gibi tanımlanır:

$$(R \cup S)(u,v) = \max\{R(u,v), S(u,v)\}$$

Örnek. Bu işlemin uygulanmasını bir örnek üzerinde anlatacağız. Varsayalım ki, yeni bir fabrikanın üretim birimleri farklı uzmanlar tarafından değerlendirilmiş ve sonuçlar ilişkilerde gösterilmiştir. Değerlendirmeler 3 parametre (birimin güvenliği, başarımı ve çevreye etkisi) ile yapılmıştır. Bu parametrelerin bulanık değerlendirilmesinde ise 3 sözel değer kullanılmıştır:

Güvenlik değişkeni için {az güvenli, orta derecede güvenli ve çok güvenli}

Başarım değişkeni için {az başarılı, orta derecede başarılı ve çok başarılı}

Çevreye etkisi için {az etkili, orta derecede etkili ve çok etkili}.

Basit olması için tüm değişkenlerin uygun olarak “az”, “orta” ve “çok” değerlerini sırasıyla C,B,A ile ifade edelim.

Uzman 1 ve Uzman 2’nin birimlerde yaptıkları değerlendirme sonuçları uygun olarak U1 ve U2 Çizelgelerinde gösterilmiştir.

Çizelge 3.1 Uzman1’in değerlendirmesi-U1 ilişkisi

Birim_no	Güvenlik	μ_g	Başarım	μ_b	Çevre	μ_c
B1	C	0.8	C	0	C	0.2
B1	B	0.3	B	0.8	B	0.4
B1	A	0	A	0.9	A	0
B2	C	0	A	1	A	0
B2	B	0.7	B	0	B	0.8
B2	A	0.6	C	0	C	0.6
B4	C	0	A	0	A	0.5
B4	B	0.6	B	0.7	B	0.7
B4	A	0.6	C	0.4	C	0

Çizelge 3.2. Uzman 2’in değerlendirmesi-U2 ilişkisi

Birim_no	Güvenlik	μ_g	Başarım	μ_b	Çevre	μ_e
B1	C	0.6	C	0	C	0.4
B1	B	0.7	B	0.9	B	0.2
B1	A	0	A	0.9	A	0
B2	C	0	C	0.8	C	0
B2	B	0.9	B	0.4	B	0.6
B2	A	0.2	A	0	A	0.8
B3	C	0.3	C	0.7	C	0.3
B3	B	0.8	B	0.7	B	0.8
B3	A	0	A	0	A	0

U1 ilişkisindeki ilk 3 satır üzerinde ilişkilerdeki özellik değerlerinin anlamını açıklayalım.

B1 birimi 0.8 üyelik derecesi ile az güvenli, 0.3 üyelik derecesi ile orta derecede güvenlidir. Birim 0.8 ile orta, 0.9 üyelik derecesi ile çok başarılıdır. Birim 0.2 derece ile az, 0.4 derece ile orta etki göstermektedir.

Sorgu: Sınamaları yapılmış tüm birimler için uzmanların değerlendirmeleri sorgusunu inceleyelim.

Bu sorgu U1 ve U2 ilişkilerinin birleşimi işlemini yapmakla gerçekleştirilir.

$$U_{12}=U_1 \cup U_2$$

Sonuç çizelge aşağıdaki gibi olacaktır:

Çizelge 3.3. U1 ve U2 ilişkilerinin birleşimi- $U_{12}=U_1 \cup U_2$

Birim_no	Güvenlik	μ_g	Başarım	μ_b	Çevre	μ_e
B1	C	0.8	C	0	C	0.4
B1	B	0.7	B	0.9	B	0.4
B1	A	0	A	0.9	A	0
B2	C	0	A	1	A	0
B2	B	0.9	B	0.4	B	0.8
B2	A	0.6	C	0	C	0.8
B4	C	0.6	A	0.4	A	0
B4	B	0.6	B	0.7	B	0.7
B4	A	0	C	0	C	0.5
B3	C	0.3	A	0.7	A	0.3
B3	B	0.8	B	0.7	B	0.8
B3	A	0	C	0	C	0

3.1.2. Kesişim ($r \cap s$)

R ve S, $X \times Y$ üzerinde ikili ilişkiler olsun. R ve S bulanık ilişkilerinin kesişimi aşağıdaki gibi tanımlanır:

$$(R \cap S)(u,v) = \min\{R(u,v), S(u,v)\}$$

Örnek olarak yine U1 ve U2 ilişkilerini kullanacağız.

Sorgu: Sınamaları tüm uzmanlar tarafından yapılmış birimler için değerlendirme sonuçlarını bulan sorguyu inceleyelim.

Bu sorgu U1 ve U2 ilişkilerinin kesişimi ile gerçekleştirilir.

Çizelge 3.4. U1 ve U2 ilişkilerinin kesişimi $U1 \cap U2$

Birim_no	Güvenlik	μ_g	Başarım	μ_b	Çevre	μ_c
B1	C	0.6	C	0	C	0.2
B1	B	0.3	B	0.8	B	0.2
B1	A	0	A	0.9	A	0
B2	C	0	A	0.8	A	0
B2	B	0.7	B	0	B	0.6
B2	A	0.2	C	0	C	0.6

3.1.3. Fark (r - s)

Birleşebilir r ve s ilişkilerinin farkı, r ilişkisinde bulunan, fakat s ilişkisinde bulunmayan satırlardan oluşan ilişkidir.

3.1.4. Kartezyen Çarpım (r X s)

$A \in F(X)$ ve $B \in F(X)$ ilişkilerinin kartezyen çarpımı aşağıdaki gibi hesaplanır:

$$(A \times B)(u,v) = \min\{A(u), B(v)\}; \text{ tüm } u \in X \text{ ve } v \in Y \text{ ler için}$$

r ve s' in kartezyen çarpımı $r \times s$, iki ilişkiden oluşturulmuş bir ilişkidir. Sonuç ilişki, r'in her bir satırının s'in her bir satırı ile birleşmesi yolu ile alınır.

Kartezyen çarpımı sonucu alınan ilişki çok büyük olabileceğinden örnek olarak U1 ve U2 ilişkilerinin alt ilişkilerini kullanacağız. U1K ve U2K ilişkilerini aşağıdaki gibi tanımlayalım:

Çizelge 3.5. Küçültülmüş U1 ilişkisi

Birim_no	Güvenlik	μ
----------	----------	-------

Çizelge 3.6. Küçültülmüş U2 ilişkisi

Birim_no	Güvenlik	μ
----------	----------	-------

B1	C	0,8
B1	B	0,3
B1	A	0

B1	A	0
B1	B	0,9
B1	C	0,9

U1K ve U2K'nın kartezyen çarpımı sonucunda alınan ilişki aşağıdaki çizelgede gösterildiği gibi olacaktır:

Çizelge 3.7. U1K ve U2K ilişkilerinin kartezyen çarpımı

Birim_no	Güvenlik	μ_g	Birim_no	Başarım	μ_b
B1	C	0,8	B1	A	0
B1	C	0,8	B1	B	0,9
B1	C	0,8	B1	C	0,9
B1	B	0,3	B1	A	0
B1	B	0,3	B1	B	0,9
B1	B	0,3	B1	C	0,9
B1	A	0,8	B1	A	0
B1	A	0,8	B1	B	0,9
B1	A	0,8	B1	C	0,9

3.1.5. İzdüşümü ($P_x(r)$)

$r(R)$, bulanık ilişki olsun ve S, R'in altkümesi olsun: $S \subset R$. O zaman r ilişkisinin S üzere izdüşümü aşağıdaki gibi ifade edilir:

$$\Pi_S(r) = \{t(S) | (\forall u) (u \in r(R) \wedge t = \cup_{f u} [R])\}$$

Burada t(S) s ilişkisinin satırlarıdır.

İzdüşümü, seçme ve bitleştirme işlemleri ile bağlı açıklamalar ve örnekler bir sonraki altbölümde verilmiştir.

3.1.6. Bitleştirme

$r(R)$ ve (S) iki bulanık ilişki olsun. $P_f, A \ominus B$ şeklinde koşul yüklemidir. Burada \ominus bulanık karşılaştırma işlemidir. O zaman bu iki ilişkinin bulanık bitişirmesi bulanık seçme işleminin yardımıyla şu şekilde ifade edilebilir:

$$r \bowtie_{P_f} s = \sigma_{P_f}(r \times s).$$

Bulanık bitişirmenin özel hali olarak bulanık doğal bitişirmeyi aşağıdaki şekildeki gibi tanımlayabiliriz.

$$r \bowtie s = \{t((R-Q) \cup S) \mid (\exists u)(\exists v)(u \in r \wedge v \in s \wedge SE_{\alpha}(u[Q], v[Q]) \geq \beta \wedge t[R-Q] = u[R-Q] \wedge t[S-Q] = v[S-Q] \wedge t[Q] = u[Q] \cap v[Q])\}$$

3.2. Benzerlik İlişkileri

Üyelik dereceleri ile ifade edilmiş bulanık veriler arasında benzerliğin, yakınlığın bulunması için benzerlik ilişkileri kullanılmaktadır. Şimdi, örnekler üzerinde benzerlik ilişkilerinin, ilişkisel cebir işlemleri ile nasıl ifade edilebileceğini ele alacağız.

Örnek. $x \in X$ ve $y \in X$ olsun. X ülkeler kümesidir. $\{x_1, x_2, x_3\}$ ve $\{y_1, y_2, y_3, y_4\}$ X kümesinin iki alt kümesi olsun: Kümeler kesişebilir. İki ilişki: ülkeler arasında yüzölçümlerinin yakınlığını ifade eden Y ilişkisini ve coğrafi olarak yakınlığı, komşuluğu ifade eden K ilişkisini tanımlayalım. O zaman Y ilişkisi “ x ülkesi ile y ülkesinin yüzölçümleri yakındır” , K ilişkisi “ x ülkesi ile k ülkesi coğrafi olarak yakındırlar” anlamlarını ifade etsin. $\{x_1, x_2, x_3\} = \{\text{Türkiye, Azerbaycan, Polonya}\}$, $\{y_1, y_2, y_3, y_4\} = \{\text{Japonya, İran, Şili, Polonya}\}$ olsun. Bu ilişkiler matris şeklinde aşağıdaki gibi ifade edilebilir:

Çizelge 3.8. Y matrisi

	Y_1	Y_2	Y_3	Y_4
X_1	0.8	0.9	0.6	0.2
X_2	0.1	0.2	0.3	0
X_3	0.8	0.7	0.5	0.1

Çizelge 3.9. K matrisi

	Y_1	Y_2	Y_3	Y_4
X_1	0	1	0	1
X_2	0	1	0	1
X_3	0	0.3	0	0.9

Y ve K ilişkilerinin kesişimi “ x’ in yüzölçümü, y’ den çok büyüktür ve x, y ’ye coğrafi olarak çok yakındır” anlamını verir. Kesişim işleminin sonucu Çizelge 3.10 ‘da gösterilmiştir.

Çizelge 3.10. Y ve K ilişkilerinin kesişimi

	Y ₁	Y ₂	Y ₃	Y ₄
X ₁	0	0.9	0	0.2
X ₂	0.1	0.2	0	0
X ₃	0	0.3	0	0.1

	Y ₁	Y ₂	Y ₃	Y ₄
X ₁	0.8	1	0.6	1
X ₂	0.1	1	0.3	1
X ₃	0.6	0.7	0.5	0.9

Çizelgedeki verilere dayanarak söyleyebiliriz ki, Polonya ve İran’ın yüzölçümleri ve coğrafi olarak birbirlerine yakınlık derecesi 0,3’tür.

Şimdi sorgumuz şu şekilde olsun: “x ’in yüzölçümü y ’ye yakındır veya x, coğrafi olarak y’ ye yakındır). Bu sorguyu bulanık birleşim işlemi ile gerçekleştireceğiz. $(R \cup S)(x,y)$ bulanık birleşim işleminin sonucu aşağıdaki gibi olacaktır:

Çizelge 3.11. Bulanık birleşim sonucu

Örneğin: “Azerbaycan’ın ve Rusya’nın yüzölçümleri aynıdır veya çok yakın komşudurlar” cümlesi 1 üyelik derecesi ile doğrudur.

İzdüşümü işlemi ile benzerlik ilişkisinin gerçekleştirmesini örnek üzerinde anlatalım. Varsayalım ki, R ilişkisi “x, y ’den yeterli derecede büyüktür” anlamını veriyor.

$$R = \begin{pmatrix} & y_1 & y_2 & y_3 & y_4 \\ x_1 & 0.8 & 0.1 & 0.1 & 0.7 \\ x_2 & 0 & 0.8 & 0 & 0 \\ x_3 & 0.9 & 1 & 0.7 & 0.8 \end{pmatrix}$$

R' in X üzere izdüşümü aşağıdaki anlamları verecek:

X₁'e (x₁,y₁), (x₁,y₂), (x₁,y₃) (x₁,y₄) satırlarından en yüksek üyelik derecesi atanıyor.

Π_x (x₁)=0.8; bu birinci satırda en yüksek derecedir.

X₂'ye (x₂,y₁), (x₂,y₂), (x₂,y₃) (x₂,y₄) satırlarından en yüksek üyelik derecesi atanır.

Π_x (x₂)=0.8; bu ikinci satırda en büyük değerdir.

X₃'e (x₃,y₁), (x₃,y₂), (x₃,y₃) (x₃,y₄) satırlarından en yüksek üyelik derecesi atanır.

Π_x (x₃)=1; bu üçüncü satırda en büyük değerdir.

Bulanık C ∈ F(X) kümesi ve bulanık R ∈ F(XxY) ilişkisinin sup-min bileşimi aşağıdaki gibi tanımlanır:

$$(C \circ R)(y) = \sup \min \{C(x), R(x, y)\} \quad \text{tüm } y \in Y \text{ için.}$$

Örnek: A ve B, x ∈ X bulanık sayılar ve R = A x B bulanık ilişki olsun.

Bileşim işleminin aşağıdaki özellikleri bulunmaktadır:

$$A \circ R = A \circ (A x B) = A,$$

$$B \circ R = B \circ (A x B) = B.$$

C , {1, 2, 3} evreninde bulanık kümedir ve R, {1, 2, 3} kümesinde ikili bulanık ilişkidir. Varsayalım ki, C = 0.2/1 + 1/2 + 0.2/3 ve R ilişkisi aşağıdaki gibidir:

Çizelge 3.12. R ilişkisi

	1	2	3
1	1	0.8	0.3
2	0.8	0	0.8
3	0.3	0.8	0

sup-min bileşeninin tanımını kullanmakla aşağıdaki sonucu elde ederiz:

$$C \circ R = (0.2/1 + 1/2 + 0.2/3) \circ R = 0.8/1 + 1/2 + 0.8/3$$

Şimdi sup-min işleminde her iki bileşenin ilişki olduğu duruma bakalım.

$$R \in F(X \times Y) \text{ and } S \in F(Y \times Z)$$

R ve S'in sup min bileşimi $R \circ S$ aşağıdaki gibi tanımlanır:

$$(R \circ S)(u,w) = \sup \min \{R(u, v), S(v,w)\}, v \in Y$$

Aşikârdır ki, $R \circ S$, $X \times Z$ üzerinde ikili bulanık ilişkidir.

Örnek. R ve S bulanık ilişkiler olsun.

R="x, y' den çok büyüktür.";

S = "y, z'ye çok yakındır"

R=

	Y ₁	Y ₂	Y ₃	Y ₄
X ₁	0.8	0.1	0.1	0.7
X ₂	0	0.8	0	0
x ₃	0.9	1	0.7	0.8

S =

	z ₁	z ₂	z ₃
Y ₁	0.4	0.9	0.3
Y ₂	0	0.4	0
y ₃	0.9	0.8	0.8
y ₃	0.6	0.7	0.5

R ve S' nin bileşimi aşağıdaki gibi olacaktır:

R ∘ S=

	z ₁	z ₂	z ₃
X ₁	0.6	0.8	0.5
X ₂	0	0.4	0
x ₃	0.7	0.9	0.9

R ve S ilişkilerinin bileşimi, R ve S matrisleri arasındaki klasik çarpı işlemidir, fark yalnızca toplama yerine maksimum; çarpma yerine ise minimum işleminin kullanılmasındadır.

Şimdi ilişkiler arasında anlamsal yakınlığın-benzerliğin belirlenmesi için gereken birkaç yeni kavramı öğrenelim. Bunlar anlamsal uzay (semantic space), anlamsal eşdeğerlilik (dercesi-semantic equalency degree) ve anlamsal eşitlik (semantic equalence) kavramlarıdır.

Kesin veriler evrende nokta ile bulanık veriler ise uzayda noktalar kümesi ile ifade edilir. Bulanık verinin anlamı, anlamsal uzay denen alana uygun mümkünlük dağılımı ile ifade edilir. İki bulanık veri arasındaki anlamsal ilişki, bu ilişkilerin anlamsal uzayları arasındaki ilişki ile ifade edilebilir. SS (π_A) ve SS (π_B) uygun olarak π_A ve π_B bulanık

verilerinin anlamsal uzayları olsun. Eğer $SS(\pi_A) \supseteq SS(\pi_B)$ veya $SS(\pi_B) \supseteq SS(\pi_A)$ ise o zaman diyebiliriz ki, π_A , π_B 'yi anlamsal olarak kapsar veya π_B anlamsal olarak π_A 'yi kapsar. Eğer $SS(\pi_A) \supseteq SS(\pi_B)$ ve $SS(\pi_B) \supseteq SS(\pi_A)$ ise π_B ve π_A ilişkilerinin anlamsal olarak eşdeğer olduğunu söyleyebiliriz. İki π_B ve π_A ilişkileri arasındaki anlamsal eşdeğerlik derecesi $SID(\pi_A, \pi_B)$ şu şekilde tanımlanabilir:

$$SID(\pi_A, \pi_B) = (SS(\pi_B) \cap SS(\pi_A)) / SS(\pi_B)$$

$SID(\pi_A, \pi_B)$ ifadesi, π_B 'nin anlamsal uzayının hangi yüzdesinin tümüyle π_A anlamsal uzayı tarafından kapsanmış olduğu anlamını verir.

π_A ve π_B ilişkilerinin eşdeğerlik derecesi bu ilişkiler arasındaki anlamsal eşdeğerlik dereceleri ile ifade edilebilir:

$$SE(\pi_A, \pi_B) = \min(SID(\pi_A, \pi_B), SID(\pi_B, \pi_A))$$

$U = \{u_1, u_2, \dots, u_n\}$ evreninde $\pi_A(u_i)$, $u_i \in U$, u_i 'nin doğru olması mümkünliğini ifade etsin. Yakınlık ilişkisinde α eşik değerini dikkate alırsak, $SID(\pi_A, \pi_B)$ anlamsal eşdeğerlik derecesini bulabiliriz.

$$SID(\pi_A, \pi_B) = \sum_{i=1}^n \min(\pi_B(u_i), \pi_A(u_i)) / \sum_{i=1}^n (\pi_B(u_i))$$

$$\text{Örnek: } n_1 = \{1.0/a, 0.95/b, 0.9/c\} \text{ ve } n_2 = \{0.95/a, 0.9/b, 1.0/d, 0.3/e\}$$

$U = \{a, b, c, d, e, f\}$ alanında iki bulanık veri olsun. U üzerinde oluşturulmuş yakınlık ilişkisinde α eşik değerini 0.9'a eşit kabul edelim.

O zaman:

$$SID((\pi_1, \pi_2) = \{0.95 + 0.9 + 0.9\} / \{0.95 + 0.9 + 1.0 + 0.3\} = 0.873,$$

$$SID((\pi_2, \pi_1) = \{0.95 + 0.9 + 0.9\} / \{1.0 + 0.95 + 0.9\} = 0.965$$

Buradan

$$SE((\pi_1, \pi_2) = \min(SID(\pi_1, \pi_2), SID(\pi_2, \pi_1)) = \min(0.873, 0.965) = 0.873 \text{ alırız.}$$

Özellik değerinin sayısal küme şeklinde ifadesi edildiği ve ilişkiler arasında benzerlik (yakınlık) derecelerinin dikkate alınmasını gerektiren bir örnek üzerinde de ilişkişel cebir işlemlerini anlatalım.

Bu örnekte bulanık özelliğin değeri küme şeklinde verilmiştir. Örneğimizdeki ilişkiler lisansüstü öğrenciler hakkında bilgileri içeriyor. r ve s ilişkilerinde öğrencinin numarası, okuduğu bölüm ve yaşı hakkında bilgiler bulunmaktadır (Zongmin, 2005).

Çizelge 3.13. r bulanık ilişkisi

Öğrenci_No	Bölüm	Yaş
------------	-------	-----

S9106	BM	{0.3/19, 0.8/20, 0.7/21}
S9107	BM	{0.6/30, 0.9/31, 0.7/32}
S9711	BB	{0.5/32, 0.7/33, 0.6/34}

Çizelge 3.14. s bulanık ilişkisi

Öğrenci_No	Bölüm	Yaş
S9106	BM	{0.8/20, 0.7/21}
S9108	BM	{0.5/30, 0.9/31, 0.7/32}
S9711	BB	{0.8/32, 0.7/33, 0.7/34}

Yaş özelliği üzere yakınlık ilişkisinin aşağıdaki gibi olduğunu varsayalım.

Çizelge 3.15. Yaş özelliği üzere yakınlık ilişkisi

	19	20	21	30	31	32	33	34
19	1	0.7	0.6	0	0	0	0	0
20	0.7	1	0.7	0	0	0	0	0
21	0.6	0.7	1	0.1	0	0	0	0
30	0	0	0.1	1	0.5	0.4	0.3	0.2
31				0.5	1	0.6	0.5	0.4
32				0.4	0.6	1	0.7	0.6
33				0.3	0.5	0.7	1	0.9
34				0.2	0.4	0.6	0.9	1

r ve s ilişkilerinin birleşimi sonucu Çizelge 3.16'da gösterilmiştir.

Çizelge 3.16. r ve s ilişkilerinin birleşimi $r \cup s$

Öğrenci_No	Bölüm	Yaş
S9106	BM	{0.3/19, 0.8/20, 0.7/21}
S9107	BM	{0.6/30, 0.9/31, 0.7/32}
S9711	BB	{0.8/32, 0.6/33, 0.7/34}
S9108	BM	{0.6/30, 0.9/31, 0.7/32}

r ve s ilişkilerinin farkı r-s aşağıdaki gibi bir ilişki olacak:

Çizelge 3.17. r ve s ilişkilerinin farkı

Öğrenci_No	Bölüm	Yaş
S9106	BM	{0.3/19}
S9107	BM	{0.6/30, 0.9/31, 0.7/32}
S9711	BB	{0.7/33}

3.2.1. İzdüşümü

S ilişkisi aşağıdaki gibi olsun:

Çizelge 3.18. S ilişkisi

Öğr_no	Bölüm	Yaş	Derece
9106	BM	{0.5/22,0.9/23,0.7/24}	Y.Lisans
9207	BM	{0.4/26,0.8/27,0.6/28}	Doktora
8207	BS	{0.2/24,0.9/25, 1/26}	Doktora
8108	BS	{0.1/28,0.7/29,0.5/30}	Y.Lisans
8109	BS	{0.6/29,0.7/30,0.1/31}	Y.Lisans

S ilişkisinin (bölüm_no) ve Yaş özellikleri üzere izdüşümü sonucu aşağıdaki gibi olacaktır

Çizelge 3.18. S ilişkisinin izdüşümü

Bölüm	Yaş
BM	{0.5/22,0.9/23,0.7/24}
BM	{0.4/26,0.8/27,0.6/28}
BS	{0.2/24,0.9/25, 1/26}
BS	{0.1/28,0.7/29,0.7/30,0.1/31}

İzdüşümü işlemi sonucunda veri fazlalığı varsa aradan kaldırılır. S ilişkisinin 1, 2 ve 3. satırlarında veri fazlalığı yoktur. 4 ve 5. satırlarda ise veri fazlalığı bulunmaktadır: [0.7/29 ve 0.6/29] ve [0.5/30 ve 0.7/30]. Bu iki satır üzerinde birleştirme işlemi yapmakla veri fazlalığı aradan kaldırılmıştır.

3.2.2. Seçme

P_f koşulu için r ilişkisi üzere seçme işlemi $\sigma_{P_f}(r) = \{t \mid t \in r \wedge P_f(t)\}$ şekilde tanımlanır. r ilişkisi aşağıdaki gibi olsun:

Çizelge 3.19. r ilişkisi

Kimlik_no	Bölüm	Yaş	Oda
9106	BM	{0.5/18,0.9/19,0.7/20}	Y1101
9107	BM	{0.4/20,0.8/21,0.6/22}	Y1101
9705	BS	{0.2/24,0.9/25,0.1/26}	B6280

Sorgu. Yaşı 25-26 arasında olan Bilgisayar Sistemleri (BS) Bölümü'nde okuyan lisansüstü öğrenciler hakkında bilgi veren sorguyu inceleyelim. "Yaklaşık 25-26" yaş değerinin $\{1/25, 0.26, 0.3/27\}$ mümkünlük dağılımı ile gösterildiğini varsayalım. O zaman $SEL_0(t_4(\text{Yaş}), \{1.0/25, 0.9/26, 0.3/27\}) = \min(0.82, 0.86) = 0.82 > \beta$, buradan $t(\text{Yaş} \approx_{\beta} \{1.0/25, 0.9/26, 0.3/27\})$

Görüldüğü gibi yalnız üçüncü satır bu koşulu sağlamaktadır. O zaman;

$\sigma_{\text{bölüm}=\text{BS}} \wedge \text{Yaş} = \{1/25, 0.26, 0.3/27\}$ seçme işleminin sonucu aşağıdaki gibi olacaktır:

Çizelge 3.20. Seçme işlemi sonucu

Kimlik_no	Bölüm	Yaş	Oda
9705	BS	{0.2/24,0.9/25,0.1/26}	B6280

3.1.6. Doğal Bitiştirme

Çizelge 3.21. r ilişkisi

Adı	Oda	Yaş
Ayşe	Y1101	{0.9/19,0.7/20,0.2/21}
Mehmet	Y1101	{1.0/21,0.7/22}
Ahmet	B6280	{0.6/24,1.0/25,0.7/26}
Cem	B6280	{0.7/32,1.0/33,0.7/34}

Çizelge 3.22. s ilişkisi

Kimlik_no	Bölüm	Yaş
9106	BB	{0.3/19,0.8/20,0.7/21}
9107	EE	{0.8/20,0.7/21}
9711	BS	{0.6/27,0.9/28,0.7/29}
9712	EE	{0.8/32,0.9/33,0.6/34}

r ve s ilişkileri üzerinde doğal bitiştirme (Naturel Join) işleminin sonucu aşağıdaki gibi olur. (Çizelge 3.23)

Çizelge 3.23. r ve s ilişkilerini doğal bitiştirilmesi sonucu

Kimlik_no	Adı	Bölüm	Oda	Yaş
9107	Mehmet	EE	Y1101	{0.8/21,0.7/22}
9712	Cem	EE	B6280	{0.7/32,0.9/33,0.6/34}

Bu sonucun nasıl alındığını anlatalım. Farklı yaş değerlerinin birbirine yakınlığını ifade eden bir “yaş_yakınlık” matrisimiz olsun. Bu matrisin satır ve sütunları r ve s ilişkilerinde yaş özelliğinde bulunan yaş değerleri olsun. İlişkinin hücreleri ise yaş değerlerinin birbirine yakınlığını ifade eden sayılar- bulanık değerler olsun.

$\alpha_{\text{kesim}}=0.8$ kabul edelim. O zaman anlamsal eşdeğerlilik matrisine göre ilişkilerin yalnız ikinci ve dördüncü satırlarındaki yaş özelliklerinin değerleri yaklaşık eşit olacaktır.

$$r_2[Yaş] \approx_{0.8} s_2 [Yaş] \text{ ve}$$

$$r_4 [Yaş] \approx_{0.8} s_4 [Yaş].$$

Sonuç ilişkide de yalnız bu satırlar bitleştirilmiş olacaktır. Diğer satır kombinasyonları ise ilişkide bulunmayacaktır.

İlişkisel cebirin öğrendiğimiz bulanık işlemleri, bulanık sorgu dillerinin, o sıradan SQL diline dayalı dillerin temelini oluşturmaktadır. Bulanık SQL konusu tezin sonraki bölümlerinde araştırılacaktır.

BÖLÜM 4

BULANIK VERİ TABANLARI

4.1. Bulanık Veritabanlarının Geliştirilmesi Yöntemleri

Hiyerarşik ve ağ modelleri üzerinde geliştirilmeye başlamış veritabanları sistemleri ilişkisel veri modeli ile yeni bir gelişme evrenine geçmiştir. Daha sonraları nesneye yönelik ve nesne ilişkisel modelleri geliştirilmiştir. Bu modeller çoklu ortam verileri gibi çok karmaşık veri yapılarını da ifade etme ve işleme imkânı sağlar; ama gerçek dünya uygulamaları çok daha karmaşık verilerin tasvirini ve işlenilmesini gerektirir. Klasik veri modelleri bunları gerçekleştirmekte yetersiz kalmaktadır. Özellikle, kesin, tam olmayan bilgilerin işlenilmesi çözülmesi gereken bir sorun olarak karşıya çıkmaktadır. Bu sorunun çözümü için en etkili ve yaygın yaklaşımlardan birisi bulanık veri tabanlarının geliştirilmesidir.

İlk bulanık veritabanı uygulamalarına geçen asrın 70. yıllarının sonlarında rastlanmıştır. Geçen 40 yıl içinde bulanık veri tabanları ve bulanık bilgi işlemlerinde önemli gelişmeler sağlanmıştır. Tüm veritabanı modelleri; ağ, ilişkisel, varlık –ilişkisel ve nesneye yönelik modeller bulanık kümeleri kullanarak, bulanık işlemleri gerçekleştirebilme yönünde genişletilmiştir. Günümüzde, coğrafi bilgi sistemleri ve anlamsal web alanlarında bulanık veritabanı uygulamaları ilgi çekmektedir.

Kaynaklarda bulanık ilişkisel veri tabanları (BİVT) ile bağlı araştırmalar; bulanık veri tabanlarının modelleştirilmesi ve ifade edilmesi, anlamsal ölçmeler, veri bağımsızlığı, sorguların ve verilerin işlenilmesi, veri tabanlarının çalıştırılması ve bu gibi meseleleri kapsamaktadır (Zongmin Ma, 2004).

İlişkisel modellerin esas yetersizliklerinden birisi de belirsiz ve tam olmayan verileri işleyememesidir. Var olan veritabanı modellerini, bulanık kümeleri ve bulanık mantığı kullanmakla genişletme fikri 1990'lardan itibaren uygulanmaya başlamıştır. Buckles-Petry modeli, ilişkisel modele benzerlik ilişkilerini ilave eden ilk modeldir. Bu modelde kesin olmayan bilgileri ilişkisel veri tablolarında ifade etmek için özel bir yapı sunulur. Zvieli ve Chen, varlık ilişkisel modelle bulanık mantığın birleştirilmesi yaklaşımını önermişlerdir. Onların modeli varlıkların ve ilişkilerin bulanık özelliklerini ifade etme imkânı sağlar (Zvieli ve Chen, 1986).

GEFRED (Generalized Model of Fuzzy Relational Databases) modeli bulanık alanlar üretmek için olasılığa dayalı bir modeldir. Model, bulanık verileri işlemek için esnek olanaklar sunabilmektedir. Chaudhry, Moyne ve Rundensteiner varlık-ilişki modelini genişletmekle bulanık ilişkisel veritabanlarının tasarımı yöntemini geliştirmişlerdir. Galindo, Urrutia ve Piattini genişlenmiş bulanık varlık-ilişki modelini ilişkisel veritabanlarını tasarlamak için kullanmışlardır. Bu yaklaşımda genişletilmiş varlık ilişkisel modelin bazı anlamsal yönleri ve bulanıklığı ifade etme özellikleri ortaya konulmuştur (Urrutia ve Pavesi, 2004).

Başka bir makalede ilişkisel veri tabanlarında bulanık bilgilerin tasviri için model önerilmiş ve SQL tabanlı bulanık sorgu dili- PFSQL’in özellikleri anlatılmıştır. Önerilen sistemin mimarisi şekil 4.1 ‘de verilmiştir. Sistemin merkezi bulanık sürücüye dayanmaktadır. Bu sürücü PFSQL işlemlerini gerçekleştirir ve JDBC API yazılımını çalıştırır (Galindo, 2006).

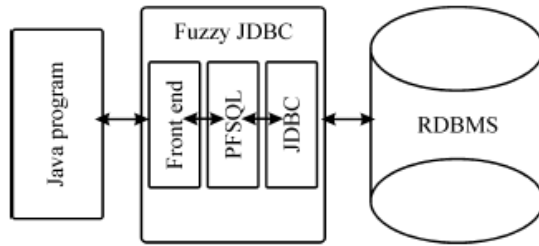
Java program bulanık JDBC tarafından önerilen ara yüzlerini “front end” bileşeni gibi kullanır. Bu ara yüzler:

Sürücü sınıflarına başlangıç değerler veriyor.

Veritabanı bağlantılarını oluşturuyor.

PFSQL cümlelerini oluşturur ve yürütür.

Sonuç kümesini okur.



Şekil 4.1. PFSQL sisteminin mimarisi (Galindo, 2006).

PFSQL cümleleri yürütüldükten sonra JDBC sürücüsünü kullanır ve sıradan SQL cümleleri gibi veritabanına gönderilir. Veritabanından alınan sonuçlar, üyelik derecelerinin ilave edilmesiyle yeniden işlenilir ve “ön son-front end” sınıfları kullanılarak Java programa geri gönderilir. PFSQL mekanizmalı JDBC sürücüsü ve FRDB modeli, veritabanı uygulamalarını geliştirmek için tam çözüm önerir. Bulanık özellikleri bulunan

veri modellerini geliştirmek için CASE aracı kullanılır. CASE aracı bulanık meta veriler aracılığıyla bulanık ilişkisel veri modeli için DDL (veri tanımlama dili) kodları üretir. CASE aracının desteklediği bulanık ilişkisel tablolar sıradan özellikleri ve bulanık özellikleri bulunan veri türlerini içerir.

Kaynaklarda, sözsel niceleyiciler kullanan bulanık mantığa dayalı veritabanı sorgulama sistemi önerilmiştir. İlişkisel veri tabanları için bulanık sorgu dili önerilmiştir.

SQLf veritabanı sorgulama dili bulanık sorgulamaları gerçekleştirmek için geliştirilmiştir. Seçme, birleştirme ve izdüşümü işlemleri bulanık koşulları da kapsayacak şekilde genişletilmiştir. Bulanık sorguların ve yuvalı bulanık SQLf sorgularının etkili işlenilmesi yöntemleri de araştırılmıştır.

Mantıksal veri tabanlarının tasarımında bütünlük sınırlamaları önemli rol oynamaktadır. Bu sınırlamalar içerisinde veri bağımlılığı önemli yer tutmaktadır. Bulanık ilişkisel veri modelleri için, bulanık işlevsel bağımlılıklar ve bulanık çok değerli işlevsel bağımlılıklarla ilgili araştırmalar yapılmıştır. Bulanık veri bağımlılıkları; verilerin yönetiminde, veri fazlalığının aradan kaldırılmasında, bulanık verilerin sıklaştırılmasında kullanılabilir.

Güncelleme anormallikleri ve veri fazlalığı sorunlarını çözmek için, ilişkisel veri tabanlarının teori yapısı, bulanık sorguları da işleyebilme yönünde genişletilmiştir. Anahtar kavramlar ve normalleştirme biçimleri de bu açıklamaya uygun olarak tanımlanmıştır. Sonuç olarak; q-anahtar, bulanık birinci normal biçim, bulanık ikinci normal biçim, q-bulanık üçüncü normal biçim ve q-bulanık Boyce-Codd normal biçimleri belirlenmiştir (Ma ve Yan, 2008).

Bulanık veri tabanlarının geliştirilmesinin bir yönü de nesneye yönelik veri tabanları ile bağlıdır. Bulanık nesneye yönelik veri tabanlarında (BNYVT) bulanıklık, nesne örnekleri seviyesinde ve sınıf hiyerarşilerinde yansıtılır. Bazı BNYVT modellerinde belirli ağırlıkları olan bulanık özellik değerleri ve SQL veri işlem dili kullanır. Bulanıklık ve belirsizliği ifade etmek için UFO (nesneye-yönelikte belirsizlik ve bulanıklık - uncertainty and fuzziness in an object-oriented) veri modeli önerilmiştir. Bu modelde nesnenin davranışları ve özellikleri kesin olmayarak ifade edilmiştir (Gysegem, Caluwe ve Vandenberghe, 1993).

Bir başka yaklaşımda bulanık veri tabanları modelleri, ağ tabanlı modeller üzerinde geliştirilmiştir. Varlıklar arasında benzerlik ilişkilerine dayanan modeller de mevcuttur.

Özet olarak veri tabanları modellerinde bulanık bilgilerin kullanılması, veritabanı araştırmalarında önemli noktalardan biridir. Şunu söyleyebiliriz ki; bulanık ve kesin olmayan bilgiler çağdaş veri uygulamaları ve bilgi sistemleri uygulamalarında yaygın olarak kullanılmaktadır. Bu bakımdan veritabanlarında bulanık verilerin sunulması ve işlenmesi konuları günceldir. Bulanık veritabanı modelleri genellikle, ilişkisel model üzerinde geliştirilmektedir; fakat klasik ilişkisel modellerin belli yetersizlikleri her türlü bilgiyi, bulanık ve kesin olmayan verileri işleme imkânı sağlayamamaktadır. Bu anlamda nesneye yönelik veritabanı modeli, daha karmaşık nesnelere ve onlar arasındaki hiyerarşik bağlantıları ifade etmekle, bulanık işlemler için daha etkili bir model olabilir. Günümüzdeki araştırmalar, daha karmaşık nesnelere ve kesin olmayan, belirsiz bilgileri birlikte işlemek için nesneye yönelik veri tabanlarını genişletmek yönündedir.

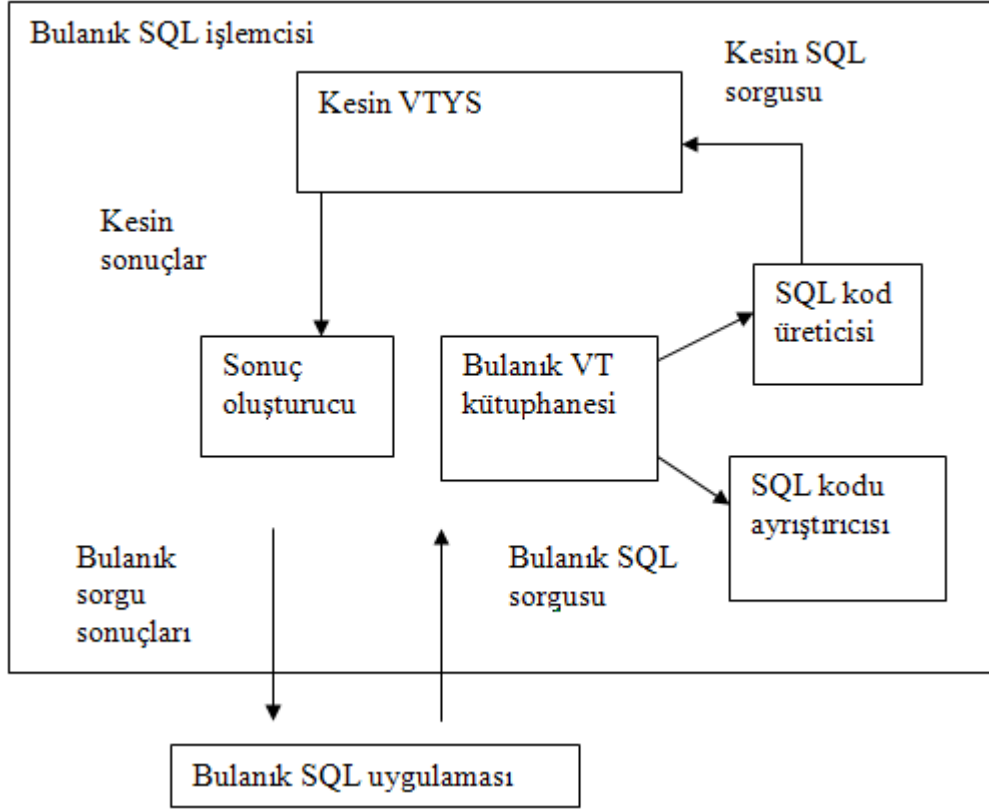
FSQL Server Bulanık İlişkisel Veritabanları için geliştirilmiştir (Bosc ve Pivert 1995). FSQL dili (Fuzzy SQL) SQL dilinin genişlemesidir. Fuzzy SQL bulanık sorguları ifade etmeye imkân verir. Fuzzy SQL için sunucu, Oracle VTYS üzerinde oluşturulmuştur. FSQL Server, klasik ve bulanık verileri tablolara kaydetme ve üzerinde işlemler yapma imkânı sağlar. Özelliklerin bulanık nitelik taşıdığını ifade etmek için sözsöz etiketler kullanılır. Klasik ilişkisel cebirin karşılaştırma işlemleri (=,>,< gibi) yanı sıra FSQL bulanık karşılaştırma işlemleri de içerir.

Pek çok gerçek dünya sorunu, kesin bilgilerden daha fazla bulanık ve kesin olmayan bilgileri içerir. Veri tabanlarında çağdaş gelişmeler, gerçek dünya sorunlarının çözümünde bulanık ve belirsiz bilgileri kullanabilmek için bulanık kümelerden yararlanmaktır. Çoklu veritabanları sistemlerinde sorgu, işlenmesi yeterli seviyede araştırılırsa da, bulanık çoklu veri tabanları için bunu söylemek doğru olmaz. Bulanık satırlar kaynağı: FTS ilişkisel modeli esasında bulanık çoklu veritabanları için sorgu işleme dilinin geliştirilmesi sorunları incelenmiş, SQL'in geliştirilmesi için bulanık SQL dili önerilmiştir. Bu dil dağıtık veritabanlarında bulanık sorguların işlenmesini sağlar.

4.2. Bulanık Veritabanı Uygulamaları

Bu bölümde bulanık veritabanları uygulamalarının ve yazılım sistemlerinin farklı alanlarda kullanımı örnekleri incelenecektir. Bu tür sistemlerden birisi OMRON sistemidir.

OMRON sistemi, Oracle VTYS (Veritabanı Yönetim Sistemi) için arayüzü olarak geliştirilmiştir. Bu sistem, bulanık veritabanı imkânlarını destekleyen sistem bulanık veritabanı uygulama programlarından, bulanık SQL işlemcisinden ve klasik veritabanı sisteminde oluşur. Bulanık veritabanı uygulama programı makine dilinde ve bulanık SQL’de yazılmıştır. Bulanık SQL işlemcisi Oracle V7 sisteminin sunucu işlemcisi gibi tasarlanmıştır. OMRON bulanık veritabanının mimarisi şekil 4.2. ‘de verilmiştir.



Şekil 4.2. OMRON Bulanık Veritabanı mimarisi

FQUERY sistemi, küçük veritabanı sistemlerinde bulanık sorgulama işlemlerini gerçekleştirmek için geliştirilmiştir. Sistemin başlıca hedefi; IBM PC veya diğer bireysel bilgisayarlarda VTYS sistemleri üzerinde ticari uygulamadır. Genel olarak işleyişi OMRON’a benzerdir. Sistem SQL’in genişlenmesi, veri tasvirleri, kullanıcı arayüzü gibi bileşenleri içeriyor. Bulanık SQL sorguları, bilgisayarda kurulu VTYS’in, örneğin Microsoft Access’in kullanabileceği biçime dönüştürülür.

Bulanık veritabanları özel amaçlar için oluşturulmuş sistemlerde de kullanılmaktadır. Böyle sistemlerden birisi, Datacycle veritabanı sistemidir. Bu sistem, tüm

veritabanında etkili, geniş arama yapmak için filtreleme teknolojisini kullanır. Sistemde sorguların işlenmesi bulanık yüklemelerin kullanılmasına dayanıyor.

Bir başka veritabanı uygulaması Birleşmiş Devletlerin posta hizmeti için geliştirilmiş DPES sistemidir. Sistemde, adres hataları ile ilgili sorunları gidermek ve mektupların doğru adrese yöneltilmesini sağlamak için bulanık veritabanı teorisi uygulanmaktadır.

Decision Plus sistemi, programların yazılması ve derlenmesi için Windows tabanlı geliştirme ortamı sağlar. Sistemde giriş ve çıkış değişkenleri arasındaki ilişki, bulanık kurallarla ifade edilir.

AIS Fuzzy Server sistemi, “Akıllı Sorgulama Sistemi” geliştirmek için bir araçtır. Sistemin temelini bulanık mantık oluşturur.

Özetlememizden de anlaşıldığı gibi bulanık veritabanları modelleri iki yönde geliştirilmiştir:

1. İlişkisel veritabanı modeli üzerinde.
2. Nesneye yönelik veritabanı modeli üzerinde.

İlişkisel veri modeli, veritabanlarına bulanık kümeleri ilave etmek için yapılan çalışmalarda esas yeri tutar. Bu çalışmaların pek çoğunda temel ilişkisel model ve sorgu dilleri, kesin olmayan verileri tasvir etmek ve veritabanından çıkarabilmek için genişletilmiştir. İlişkisel veri modeline bulanıklık elementlerinin dâhil edilmesi için iki genel yaklaşım mevcuttur. Bunlardan birincisi; alan verileri arasında klasik eşitlik kavramını benzerlik, yakınlık, farklılık kavramları ile değiştirme fikrine dayanmaktadır. İkinci yaklaşım ise özellik değerleri için doğrudan mümkünlik dağılımının kullanılmasına dayanır.

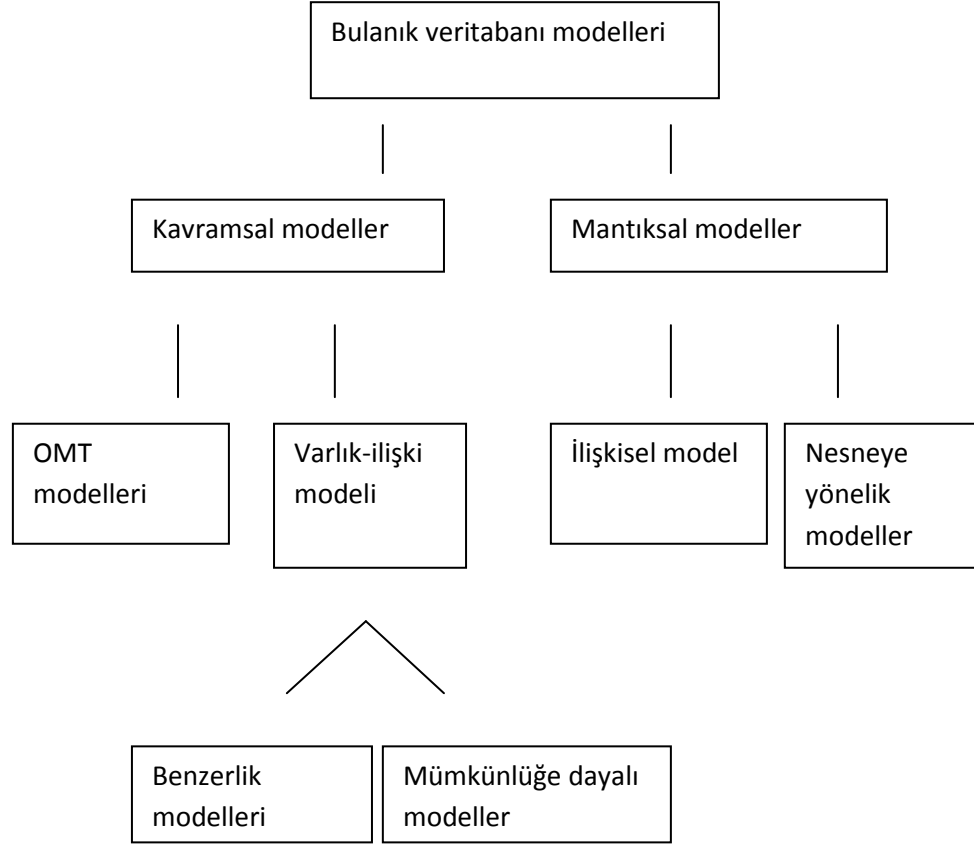
Her iki türü birleştiren karışık yaklaşım da uygulamalarda yer bulmuştur.

Kavramsal bulanık veri modelleri ise genellikle iki yönde geliştirilmektedir:

1. Varlık-ilişki modelinin bulanık kavramlarla genişletilmesi,
2. Nesneye yönelik yöntemlerin (OMT) genişletilmesi.

Bulanık veritabanı modellerinin geliştirilmesi yönleri şekil 4.3 ’de gösterilmiştir.





Şekil 4.3. Bulanık veritabanı modellerinin sınıflandırılması

4.3. Bulanık İlişkisel Veritabanlarının Teori Temeli

1970 yıllarında Codd tarafından ilişkisel modelin önerilmesi ile veritabanları sistemleri teorik ve pratik anlamda hızlı gelişim gösterir. İlişkisel modele dayalı veritabanları sistemleri yalnız iyi tanımlanmış ve kesin verilerle işleme imkânı sağlar. Gerçek yaşamda ise, belirsiz ve muğlak veriler mevcuttur ve bu tür verileri klasik ilişkisel modelle tanımlamak mümkün değildir. Günlük yaşamımızda çoğu zaman bulanık verilere dayalı kararlar vermek zorunda kalırız. Bu bakımdan, bulanık verileri ifade edebilen ve işleyebilen veritabanları sistemlerinin geliştirilmesi uygulamalarda büyük önem taşımaktadır. Bulanık ilişkisel veritabanları, Zadeh'in önerdiği mümkünlük dağılımı fonksiyonuna $\mu_{A(x)}$ dayanmaktadır. Eğer $\mu_{A(x)}$ yalnız sıfır ve bir değerlerini alırsa, bulanık olmayan fonksiyondur. Aksi halde $\mu_{A(x)}$ bulanık fonksiyondur.

Mümkünlük dağılımı fonksiyonunun, x nesnesinin A özelliği üzerinde kesin, kesin olmayan ve muğlak verileri ifade etmek için nasıl kullanıldığını inceleyelim.

Eğer U evreninde yalnız bir u_i değerimiz var ise $A(x)$ 'le ilgili veri kesindir.

$$\mu_{A(x)}(u) = 1, \text{ eğer } u = u_i \quad 0, \text{ eğer } u \neq u_i$$

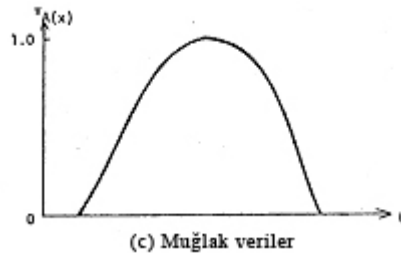
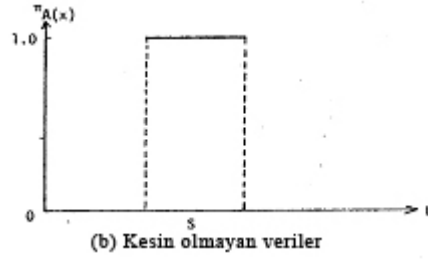
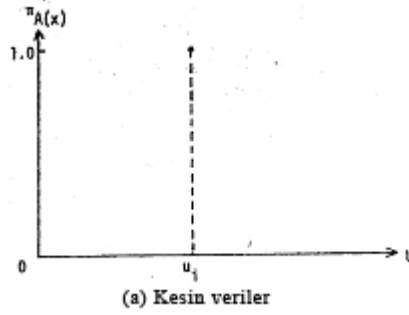
Eğer U evreninde birden fazla üyesi bulunan S bulanık olmayan kümesi var ise $A(x)$ 'le ilgili veri kesin değildir.

$$\mu_{A(x)}(u) = 1, \text{ eğer } u \in u_i$$

$$0, \text{ eğer } u \notin u_i$$

Eğer $\mu_{A(x)}$ bulanık ise $A(x)$ 'le ilgili veri muğlaktır.

Bu tanımlar esasında kesin, kesin olmayan ve muğlak verilerin grafik tasviri şekil 4,4'de verilmiştir.



Şekil 4.4. Kesin, kesin olmayan ve muğlak verilerin grafik tasviri

Kesin olmayan, muğlak verileri tanımlayabilen ve işleyebilen bulanık ilişkisel veri modeli Codd'un önerdiği ilişkisel model üzerinde oluşturulmuştur. D_f - bulanık veritabanı, R_i - genişlenmiş ilişkiler kümesidir:

$$D_f = \{R_1, R_2, \dots, R_n\}$$

R_i - genişlenmiş ilişkisi, mümkünlük dağılımlarının kartezyen çarpımının altkümesi olarak tanımlanır. Yani:

$$R_i \subseteq (P(U_{i1}) \cup \{NULL\}) \times (P(U_{i2}) \cup \{NULL\}) \times \dots \times (P(U_{im}) \cup \{NULL\})$$

Burada \times , \cup , \subseteq uygun olarak kartezyen çarpımı, birleşim işlemini ve altkümeyi ifade eder. Küme teorisinde $P(U_{ij})$, $j=1,2,\dots,m$, U_{ij} evreninde tüm mümkünlük dağılımlarının koleksiyonudur. Null ise, özellik değerinin tanımlanıp-tanımlanmadığı hakkında hiçbir bilginin olmadığı durumu ifade etmek için kullanılan özel değerdir. U_{ij} ve $P(U_{ij})$ uygun olarak temel küme ve alanı ifade eder.

KİŞİ genişletilmiş ilişkisinin AD, YAŞ ve ÇOCUK özelliklerinin olduğunu varsayalım. AD ve ÇOCUK özelliklerinin değer alanı aynıdır ve bu özellikler kişi adları kümesidir. Bunu U_1 ile ifade edelim. Yaş ise $[0,150]$ arasında sayısal değerler kümesidir. O zaman KİŞİ genişletilmiş ilişkisi

$$KİŞİ \subseteq (P(U_1) \cup \{NULL\}) \times (P(U_2) \cup \{NULL\}) \times (P(U_3) \cup \{NULL\})$$

gibi ifade edilebilir.

Çizelge 4.1 'de KİŞİ ilişkisinin bir örneği verilmiştir. Bu örneğe göre Ali 23 yaşındadır ve çocuğunun ismi Fatih'tir. Ayşe 35 yaşındadır ve Aziz ve Mehmet adlı iki çocuğu vardır. 40 yaşlı Hamit'in çocuklarının ismi Fatma veya Duygu'dur. Ayşe hakkında bilgilerin iki satırda olması, onun 2 çocuğu olmasındandır. Mahmut genç yaştadır. Çocuğunun ismi yerinde "belirsiz" yazılması, çocuğun isminin belli olmadığını gösterir. Kadir'in ise yaşı belli değil ve çocuğun ismi tanımlanmamıştır, yani çocuğu yoktur. Soner'in yaşı 50 veya 51'dir. Çocuk özelliği değerinin Null olması, çocuğun var olup olmadığı hakkında hiçbir bilginin olmadığını gösterir.

Çizelge 4.1. Bulanık ve

AD	YAŞ	ÇOCUK
Ali	23	Fatih
Ayşe	35	Aziz
Ayşe	35	Mehmet
Hamit	40	(Fatma, Duygu)

AD	YAŞ	ÇOCUK
Ali	23	Fatih
Ayşe	35	Aziz
Ayşe	35	Mehmet
Hamit	40	(Fatma, Duygu)
Mahmut	Genç	Belirsiz
Kadir	Belirsiz	Tanımlanmamış
Soner	{50,51}	NULL

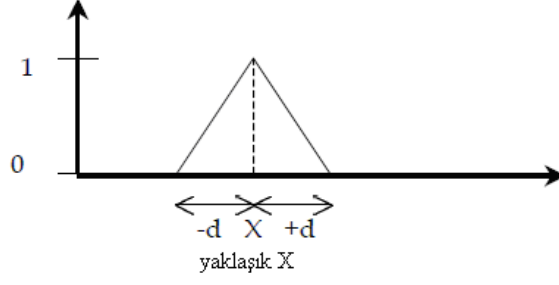
Baktığımız bu örnekte AD özelliğinin tüm değerleri kesin verilerdir, yani AD özelliğinin değer alanı kesin kümedir. YAŞ özelliği ise hem kesin, hem belirsiz, hem de bulanık değerler alabilir. YAŞ özelliğinin değer alanı sayısal ve sözel değerler içerir. ÇOCUK özelliği de kesin, bulanık ve belirsiz değerler içermesinin yanı sıra NULL değer de içermektedir. Bu ise, YAŞ ve ÇOCUK özelliklerinin değer alanlarının çok türlü olduğunu gösterir. Klasik ilişkiyel veritabanı modeli, değer alanlarının tektürlü olmasını gerektirir. Bu bakımdan YAŞ ve ÇOCUK gibi değer alanlarını çok türlü olan özellikleri klasik ilişkiyel veritabanı modelleri ile işlemek mümkün değildir.

Bulanık veritabanı modellerinde verilerin işlenebilmesini sağlamak için bu verileri sınıflandırmak gerekir.

Bulanık veritabanındaki veriler aşağıdaki gibi sınıflandırılabilir:

1. Kesin: Veride belirsizlik yoktur; örneğin, $X = 13$, Sıcaklık=25°
2. Bulanık: Bilgide belirsizlik vardır. Bu belirsizlik iki tür olabilir:
 - a. Yaklaşık değer: Bilgi tümüyle belirsiz değildir, yaklaşık bir değer vardır ve veri bu değer etrafında bir değer almaktadır, örneğin, $10 < x < 15$; Sıcaklık < 10°.

Yaklaşık değerın üçgen ifadesi şekil 4.5 'de verilmiştir.



Şekil 4.5. Yaklaşık değerin üçgen ifadesi; d, veri değerinin yayılma sınırını gösterir.

b. Sözel değişkenler: Sözel değişkenler, bulanık sayıları ifade etmekten başka, özgü bağlamda yorumlanan sözel kavramları da ifade ederler. Her bir sözel değer, fiziki anlamı olan değişken terimler (hız, ağırlık...), veya her hangi sayısal değişkenin (maaş, devamlılık, genel not ortalaması ve s) terimleri ile tanımlanabilir. Sözel değişken $\langle v, T, X, g, m \rangle$ beşlisi ile gösterilir:

Burada,

v – sözsel değişkenin adı,

T – bu sözsel değışkene uygulanan sözsel terimler,

X – X değerleri evrensel kümesi,

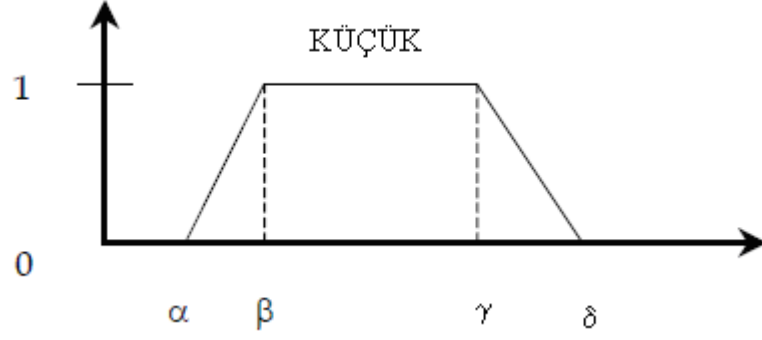
g – sözsel terimleri üretmek için gramer,

m – her bir $t \in T$ terimine atanan anlamsal kuraldır.

Sözel değışken kullandığımız zaman bilgi belirsizdir ve bu bilgiyi bulanık küme ile tanımlamaya çalışırız. Sözel terim bu bulanık kümeye verilen addır.

Örneğin; X KÜÇÜK'tür, Su SICAK'tır.

Şekilde 4,6'da KÜÇÜK sözel değışkeninin yamuk ifadesi verilmiştir.



Şekil 4.6. KÜÇÜK sözel teriminin mümkünlik dağılımının yamuk ifadesi

Burada sözel terimle bağlı 4 parametre bulunmaktadır: α , β , γ , δ . $[\beta, \gamma]$ aralığı için üyelik değeri 1.0, $[\alpha, \beta]$ ve $[\gamma, \delta]$ aralıkları için üyelik değeri $[0.0, 1.0]$ arasındadır.

4.4. Bulanık Veritabanlarının Tasviri ve Modelleştirilmesi

İlişkisel veri tabanlarında bulanık verilerin işlenebilmesini sağlayan birkaç yaklaşım mevcuttur. Bir yaklaşımda BİVT modelleri bulanık ilişkiler tabanında, diğerinde benzerlik ilişkileri tabanında oluşturulmuştur. Mümkünlük dağılımlarına dayanan modeller de vardır.

Klasik ilişkisel veri modellerinin tanımlarına uygun olarak bulanık ilişkiler tanımlanmaktadır:

$R(A_1, A_2, \dots, A_n)$ ilişkisel şeması üzerinde r bulanık ilişkisi

$Dom(A_1) \times Dom(A_2) \times \dots \times Dom(A_n)$

kartezyen çarpımının altkümesidir. Burada $Dom(A_i)$ bulanık altküme veya bulanık altkümeler kümesi olabilir.

Bulanık bilgi veya bulanık veri birkaç yolla oluşabilir:

- Ölçülen verilerde kesinliğin olmamasından dolayı,
- Öznel muhakemeler sonucu (veritabanı okulun kalitesi, evin güvenliği gibi verileri içere bilir),
- Kullanıcının istediği verinin doğasından dolayı (kullanıcı kesin olmayan bilgiler isteyebilir. Örneğin; “yazılım mühendisliği” dersinden iyi bir lisans programı”), İki yolla bulanık yaklaşımlar veri tabanlarının geliştirilmesinde kullanılabilir:
- Kesin olmayan bilgileri (verileri) depolamak ve güncellemekle,

- Özellik değerlerindeki kesinsizliğin benzerlik matrisleri ile tanımlanması yolu ile.

Çizelge 4.2. Benzerlik matrisi örneği

	Robotik	Uzman Sistemler	Yapay Zekâ	İstatistik
Robotik	1.0	0.6	0.6	0.2
Uzman sistemler	0.6	1.0	0.9	0.2
Yapay Zekâ	0.6	0.9	1.0	0.2
İstatistik	0.2	0.2	0.2	1

Özelliğin üyelik derecesi özel bir özellik olarak kaydedilir. Aşağıda verilmiş ilişkide doğal ortamlar ve bu ortamların hangi türe ait olduğunu gösteren üyelik dereceleri gösterilmiştir.

Çizelge 4.3. Özellik ve üyelik derecesi

Adı	Ortam türü	Üyelik derecesi
Akdeniz	sahil	0.9
Düzova	otlak	1
Yeşilçam	orman	0.4
Kızılkum	sahil	0.1

4.5. Bulanık Sorguların ve Verilerin İşlenilmesi

Klasik ilişkisel veri tabanları, sorguların esnekliğinden yoksundur. Sorguda verilen seçme koşulları ve içeriği kesindir. Esnek sorgular ise aşağıdaki koşulları sağlamalıdır:

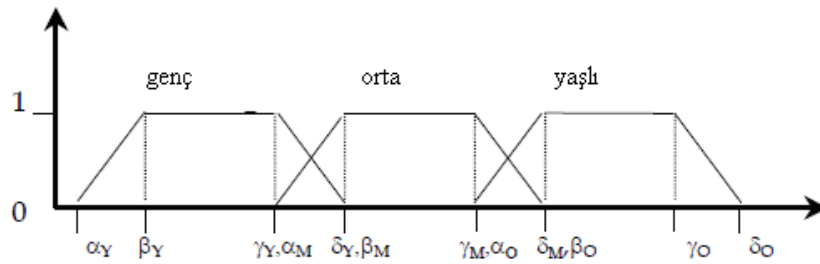
Seçilmiş satırlar arasında niteliksel farklar gösterilebilmelidir.

Eğer kullanıcı kendi bilgi ihtiyaçlarını kesin yollarla belirleyemezse belirgin olmayan koşullar kullanılabilir. Mümkündür ki; ilişkisel veritabanı tam olmayan bilgiler içerir ve sorgu koşulları kesindir veya veritabanı yalnız tam, kesin bilgiler içerse de sorgu koşulları belirgin değildir.

Kullanıcılar veritabanına kesin olmayan sorgular verebilirler. Bu sorgular:

- Kesin olmayan koşullar,
2009 yılında denetlenmiş ve geliri düşük olan tüm vergi ödeyicilerini bulmalı
- Kesin olmayan işlemler,
İhracat ve ithalat göstergileri yaklaşık aynı olan tüm ülkeleri bulmalı
- Kesin olmayan niceleyiciler kullanmakla yapılabilir.
Müşterileri çoğunlukla devlet kurumları olan şirketleri bulmalı

Veritabanlarına bulanıklığın ilave edilmesinin en kolay yolu; klasik ilişkisel veritabanlarının kullanılması ve bu veritabanlarına son kullanıcının bulanık sorgulama yapabilmesini sağlamaktır. Bulanık özelliğinin ilişkisel veritabanlarına birleştirilmesini sağlamak için özellik alanı/sözsözsel değişken üzerinde bulanık kümeler/sözsözsel terimler ilave edilir. Bunu örnek üzerinde açıklayalım: YAŞ özellik alanı üzerinde GENÇ, ORTAYAŞLI ve YAŞLI bulanık kümeleri aşağıdaki gibi tanımlanabilir (şekil 4.7).



Şekil 4.7. Genç, orta yaşlı ve yaşlı bulanık kümelerinin tanımlanması

ÖĞRENCİ çizelgesini ele alalım(Çizelge 4.4.). Bu çizelgenin Ad, Yaş, Sınıf, Genel Not Ortalaması ve Devamsızlık özellikleri bulunmaktadır.

Çizelge 4.4. Öğrenci çizelgesi

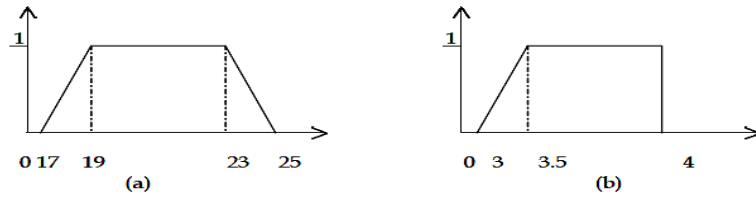
Ad	Yaş	Sınıf	GNO	Devamsızlık
Ali	19	12	83	13
Ahmet	17	10	80	9
Suzan	18	11	83	6
Fatih	19	12	56	12
Fazıl	19	12	65	32

Varsayalım ki, çok başarılı ve genç öğrencileri bulmak istiyoruz. Kesin sistem için sorgu aşağıdaki gibi olacaktır:

```
SELECT Öğrenci_adı FROM Öğrenci
```

```
WHERE (Yaş <=19) AND (Yaş <=23) AND GNO >=4
```

Ama bu sistemin çok açık yetersiz yönleri bulunmaktadır. Diyelim ki, 24 yaşlı Fatma'nın GNO'su 4'e eşittir ama o, listeye düşmeyecek, çünkü normal kesin küme mantığına göre, onun özellik değerleri verilmiş koşulu sağlamıyor. Öğrencilerin yaşlarını ve genel not ortalamalarını bulanık mantığa göre YAŞ ve GNO sözel değerleri ile ifade edelim. (şekil 4.8). O halde Fatma'nın koşulda belirlenen kümeye ait olma derecesi sıfır olmayacak, Fatma belirli bir üyelik derecesi ile bu kümeye dâhil edilecektir.



Şekil 4.8. a) YAŞ ve b) GNO sözel değerlerinin grafik ifadesi

4.6. Bulanık Veritabanlarının Çalıştırılması

Bulanık veritabanları sistemlerinin çalıştırılmasında esas işlem giriş bulanık sorgularının ayrıştırılmasıdır (Punam, Harmeet ve Malhotra, 2002). Eğer veritabanı kesin ise; yani veritabanında bulanık veriler saklanmıyorsa, INSERT sorgusu değişmeyecektir ve

ayrışmaya gerek kalmayacaktır. Bu nedenle sorgu veritabanına olduğu gibi sunulacaktır. Ayrışma sürecinde sorgu aşağıdaki maddelere ayrılır:

1. Sorgu türü: Sorgu türleri SELECT, DELETE veya UPDATE olabilir.
2. Sonuç özellikler: SELECT sorgusunun kullanımı sonucu görüntülenecek özellikler.
3. Sonuç tablolar: Sorgunun uygulandığı tablolar.
4. Koşullar: İşlem yerine getirilmeden önce belirlenecek koşullar. Bu koşullar sonradan sorgu özelliklerine (yani, sorgunun uygulanacağı özellikler) ve sözsöz terimlere bölünecektir. Eğer koşul bulanık değilse, yani sözsöz terim içermiyorsa bölünmeye gerek yoktur.

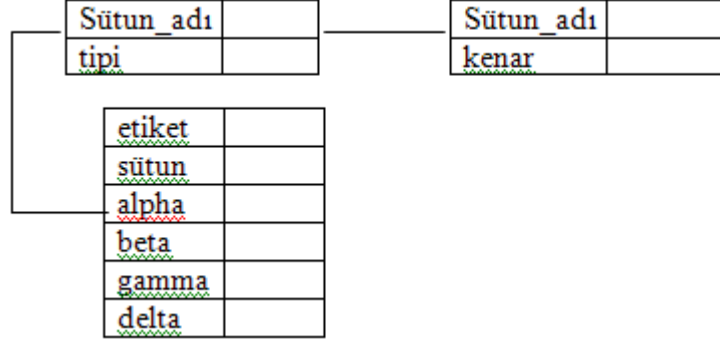
Sadece bulanık sorguların yürütüleceğini bilen bir veritabanı sistemini değil; aynı zamanda belirsiz, muğlak verilerin de tutulduğu bir veritabanı sisteminin tasarımına inceleyelim.

Varsayalım ki, yukarıda baktığımız tabloda YAŞ, YÜZDE ve DEVAMSIZLIK özellikleri bulanık değerler olabilirler. Diğer özelliklerin kesin olduğunu kabul ediyoruz.

Veritabanımızdaki özellikler iki türdür:

- . Tip 1: Özellik yalnız kesin değerler olabilir;
- . Tip 2: Özellik bulanıktır ve kesin, bulanık, yaklaşık değer veya sözsöz terim olabilir.

Veritabanlarının tasarımında veri sözlüğü veya meta verilerin oluşturulması önemli yer tutar. Bulanık veritabanındaki verileri tutmak için üç tabloya ihtiyacımız olacaktır (Punam, Harmeet ve Malhotra, 2002). Bunlar veritabanındaki özelliklerin tutulduğu VT_ÖZELLİKLERİ tablosu, yaklaşık sayıların değer sınırlarının tutulduğu YAKLAŞIK_DEĞER tablosu ve bulanık kümelerin kaydının tutulduğu ETİKETLER tablosudur (şekil 4.9).



Şekil 4.9. Bulanık veritabanının meta veri yapısı

1) VT_ SÜTUNLARI

Bu tabloda, tablodaki tüm özelliklerin türleri tutuluyor. Tablonun sütunları aşağıdakileri ifade eder:

Sütun_adı: Bu, tablonun birincil anahtarıdır ve onun satırları ÖĞRENCİLER tablosundaki özelliklere uygundur.

Tip: Bu sütunda uygun özelliklerin tipleri tutulur ve iki değer, kesin ve bulanık tiplere uygun olarak 1 ve 2 değerlerini alabilir.

SÜTUN_ADI	TİPİ
-----------	------

2) YAKLAŞIK_DEĞER_TABLOSU

Bu tabloda yaklaşık sayı (d) parametresi tutulur. Sütunların anlamı şu şekildedir:

Sütun_adı: Bu yabancı anahtardır ve VT_SÜTUNLARI tablosundaki aynı adlı sütuna uygundur.

Kenar: Bu sütun d parametresine uygundur.

SÜTUN_ADI	KENAR
-----------	-------

3. ETİKETLER

Meta veri seviyesinde ETİKETLER denen tek bir tabloya ihtiyacımız var:

ETİKET	SÜTUN_ADI	ALPHA	BETA	GAMMA	DELTA
--------	-----------	-------	------	-------	-------

Bu tablo, tüm özellik alanları üzere tanımlanmış bulanık kümelerin kaydını tutmak içindir. Tablonun her bir sütununun anlamı şu şekildedir:

. Etiket: Tablonun birincil anahtarıdır ve bulanık küme ile alakalı sözselsel terimi ifade eder.

Sütun_Adi: Bu, tabloda yabancı anahtardır ve VT_SÜTUNLARI tablosuna uygundur. Verilmiş sözselsel terimle alakalı sözselsel değişkeni ifade eder.

. Alpha, Beta, Gamma, Delta: Bulanık kümelerin sınır değerlerini - (α , β , γ , δ) ifade eder.

4.7. Bulanık Veritabanlarında Sorgu Dili

Bulanık veritabanlarında sorgu işleme ve sorgu tanımlama dili genellikle SQL dili üzerinde geliştirilmiştir. Bu bakımdan, bulanık sorgu dillerine, SQL dilinin genişlenmesi gibi bakmak mümkündür.

Bildiğimiz gibi, SQL sorgu dilinin yapısı SELECT cümlesi üzerinde oluşturulmuştur. Bulanık veritabanlarında da bu yapı korunmaktadır. Bulanık sorgu dilleri sözdizimine veya bazı terimlere göre farklılık gösterse de benzer özellikler bulunmaktadır. Burada bulanık sorgu dili SQLf üzerinde anlatılacaktır (Urrutia ve Pavesi, n.d.).

SELECT cümlesinin yapısı şu şekildedir:

SELECT <ATTRIBUTE1> [, <ATTRIBUTE2> ...]

FROM <TABLE1> [, <TABLE2> ...]

[WHERE <CONDITION1> [<CON> <CONDITION2> ...]]

[THOLD ilişkisel operator x | #x]

Burada CONDITION aşağıdaki gibi tanımlanır:

1. ATTRIBUTE ilişkisel operatör CONSTANT

2. ATTRIBUTE1 ilişkisel operatör ATTRIBUTE2 (her iki özellik karşılaştırılabilir olmalıdır).

3. ATTRIBUTE, veritabanında tanımlanmış özelliktir.

CON iki koşulu birleştirmek için kullanılan mantık işlemidir. (VE, VEYA).

THOLD sonuç kümeye uygulanan alfa_ kesimi ifade eder.

#x - geri döndürülecek kayıtlar sayısıdır.

Örnek.

SELECT AD

FROM ÖĞRENCİ

WHERE GNO > 80 AND DEVAMSIZLIK = "DÜŞÜK"

Bulanık veritabanlarında verilerin işlenmesi için klasik SQL'de olduğu gibi INSERT, DELETE, UPDATE gibi komutlar kullanılmaktadır.

İşlemlerin yapısı aşağıda verilmiştir:

1. INSERT

INSERT INTO <TABLO>

VALUES (<ifade1>, ...)

Örnek:

INSERT INTO ÖĞRENCİ

VALUES ("Akif", 19, 12, 85, 10)

2. DELETE

DELETE

FROM <TABLE>

[WHERE <CONDITION1> [<CON> <CONDITION2> ...]]

CONDITION ve CON yukarıda tanımlandığı gibidir.

Örnek:

DELETE

FROM ÖĞRENCİ

WHERE GNO > 80 AND DEVAMSIZLIK = "DÜŞÜK"

3. UPDATE

UPDATE <TABLE>

SET VALUES <ATTRIBUTE1> = <ifade1>

[, <ATTRIBUTE2> = <ifade2> ...]

[WHERE <CONDITION1> [<CON> <CONDITION2> ...]]

Burada CONDITION ve CON yukarıda tanımlandığı gibidir.

Örnek:

UPDATE STUDENTS

SET GNO = 80

WHERE GNO < 80 AND DEVAMSIZLIK = "DÜŞÜK"

SQL sorgu diline bulanıklık özelliklerinin ilave edilmesi ile gerçekleştirilmiş bulanık sorgu dilleri pek çok araştırma için konu olmuştur. Takaçi ve Skrbic (2008) makalelerinde bulanık veritabanlarında sorgulama işlemlerini gerçekleştirmek için, PFSQL sorgu dilini önermişlerdir. PFSQL dilini farklı yapan özelliği; önceliğe dayalı koşulları da işleyebilmesidir. Öncelikler genel olarak ağırlık değerleri şeklinde ifade edilir. PFSQL, FSQL sorgu dili (Urrutia ve Pavesi, n.d.) üzerinde geliştirilmiştir ve FSQL'in WHERE koşulunda belirli öncelik veya önemlilik derecesi kullanılmasını sağlar.

Bir PFSQL sorgusuna bakalım. Varsayalım ki, basket ve futbol takımlarında adaylar değerlendirilmelidir. Değerlendirme üç parametreye (*boy, hız ve fiziki güç*) göre yapılmaktadır. Spor türüne uygun olarak parametreler farklı öncelikler alabilir. Örneğin; futbolcu için *hız* çok önemlidir. *Güç* orta derecede, *boy* ise daha az derecede önem taşımaktadır. Basketçi için en önemli etken *boy*, sonra *güçtür*. *Hız* daha az önemlidir. Aşağıda örnek olarak futbolcu seçimi için PFSQL sorgusu verilmiştir.

SELECT * FROM ADAY WHERE (boy='uzunl') PR 0.2

AND (güç='çok_ iyi') PR 0.6 AND (hız='yüksek') PR

Yang, Zhang, Liu, Wu, Hiroshi ve Rische (2001) makalelerinde bulanık veritabanlarında yuvalı SQL sorgularının işlenmesi mekanizmasını önermişlerdir.

Awadhesh, Goswami ve Gupta, (2009) makalelerinde çoklu veritabanları için bulanık iş sistemini (fuzzy transaction systems) önermişlerdir. Sistem, SQL üzerinde geliştirilmiştir ve bulanık iş ilişkileri üzerinde bulanık sorguları destekler. FTS-SQL bulanık HAVING işlemini de gerçekleştirebilir.

Veri işleme ve veri ambarlama; veritabanı yöneticileri, veritabanı geliştiricileri için önemli meselelerden birisidir. Bunun çözümü için SQL Server™ 2005 veritabanı platformunda SQL Server Integration Services (SSIS) modülü önerilmiştir (Nathan, n.d.).

SSIS'in özelliklerinden birisi de veri temizleme için bulanık arama ve bulanık gruplaştırma desteğinin bulunmasıdır. Bulanıklık özelliğinin ilave edilmesi ile arama ve gruplaştırma işlemleri daha hızlı ve etkili olmuştur.

MA ve YAN (2008) makalelerinde bulanık veritabanı modellerinin özelliklerini, farklı yönlerini, geliştirilmesi yöntemlerini özetlenmişlerdir.

4.8. Bulanık Varlık İlişki Modeli

Veritabanlarının modelleştirilmesi bildiğimiz gibi iki seviyede; kavramsal veri modelleştirilmesi ve mantıksal veri modelleştirilmesi seviyelerinde gerçekleştirilmektedir. Kavramsal veri modelleri genel olarak yüksek soyutlama seviyesinde kullanılır. Bu seviye kullanıcı ile bilgi sistemi tasarımcısı arasında bir köprü rolünü oynar. Kavramsal veri modelleştirilmesinin kusursuz ve ayrıntılı yapılması, mantıksal veri modelleştirilmesi için oldukça önemlidir. Bu bakımdan, bulanık mantıksal veritabanlarının oluşturulması için kavramsal seviyede de bulanıklık kavramlarını ilave edilmelidir.

Bulanık veritabanlarının kavramsal tasarımı ile bağlı pek çok çalışma yapılmıştır. Bu çalışmalar genellikle, varlık-ilişki modeli üzerinde yoğunlaşmıştır. Zveli ve Chen tarafından önerilmiş bulanık varlık ilişki modelinde bulanıklık 3 seviyede ifade edilebilir. Bu seviyeler klasik ER modelindeki seviyelere uygundur.

Klasik ER model gerçek dünya kavramlarını varlıklar, ilişkiler ve özellikler terimleri ile tasvir eder. Varlık, ilişki ve özellikler kümeleri kesin kümelerdir.

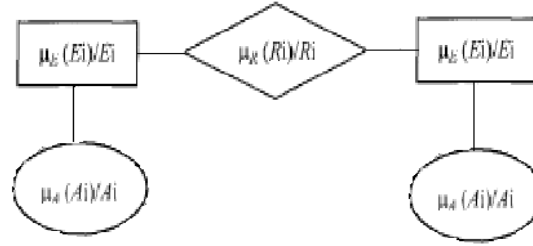
Birinci seviye varlık ve ilişki türlerini ve özellikleri kapsayan model seviyesidir.

İkinci seviye tür/örnek seviyesidir. Bu seviye varlık ve ilişki türlerini kapsar.

Üçüncü seviye özellik değerleri seviyesidir ve varlık veya özellik örneklerinin özellik değerlerini kapsar.

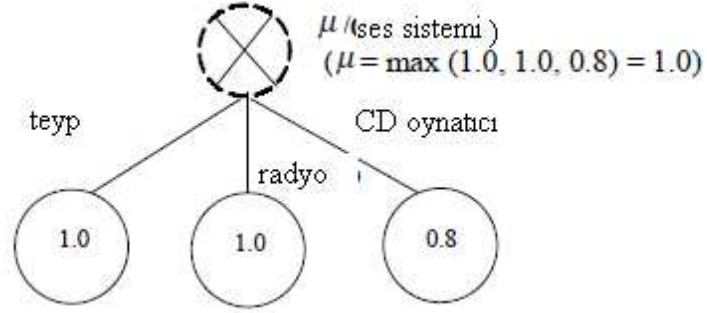
Uygun olarak ER veri modelinin bulanık genişlenmesi 3 seviyede gerçekleşmelidir. Birinci seviyede varlık türleri, ilişki türleri ve özellikler bulanık kümeler olabilir. Bu sebepten uygun kavramların üyelik fonksiyonları da modele ilave edilmelidir.

E_i ve D_i bulanık varlık türleri, A_i ve B_i özellik türleri, R_i özellik türü olsun. O zaman iki varlık arasındaki ilişkiyi gösteren bulanık ER diyagramı böyle olacak (Zongmin, 2005):



Şekil 4.10. Bulanık ER diyagramı

Genişlenmiş varlık-ilişki modelinde (EER) bulanık özelleştirme, geliştirme, kategori ve toplama kavramları kullanılmaktadır. Şekil 4.11 'de örnek olarak arabanın ses sistemi sınıfına teyp, radyo ve CD oynatıcısının ait olması bulanık toplama (aggregation) ilişkisi ile ifade edilmiştir.



Şekil 4.11. Bulanık toplama

Ma (2005) makalesinde Veri modeli için farklı bulanıklık seviyeleri önerilmiştir. Önerilen bulanık veri modelinde yazdırılabilir, soyut, özgür veri türleri, gruplaştırma, toplama, ISA ilişkisi mevcuttur. Soyut ve özgür veri türlerinde bulanıklık örnek/şema ve şema seviyelerinde gerçekleştirilir. Örnek/şema seviyesinde bulanıklık, örneğin nesneye aitlik derecesini ifade ediyor. Şema seviyesinde bulanıklık, nesnenin veri modeline uygun olması derecesini ifade ediyor.

BÖLÜM 5

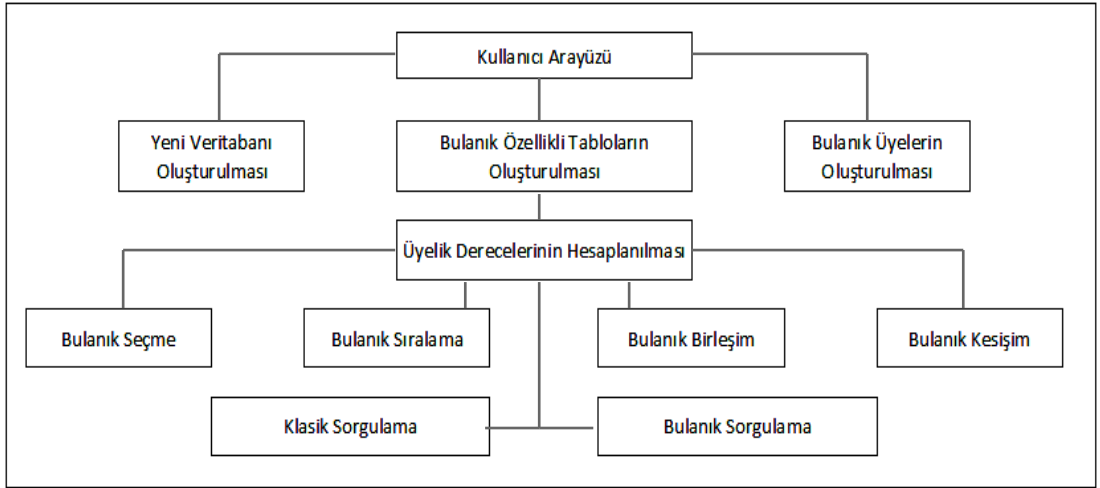
BULANIK SQL UYGULAMASI

Tezin bu bölümünde geliştirmiş olduğum Bulanık SQL uygulamasının yapısı, işleyişi ve amacı anlatılmaktadır.

Bulanık SQL uygulaması Microsoft Visual Basic 6.0 yazılım geliştirme aracı ve MS Access Veritabanı Yönetim Sistemi kullanılarak geliştirilmiştir. Geliştirmiş olduğum uygulama aşağıdaki işlemleri yerine getirmektedir:

- Bulanık tipli özellikler de içerebilen veritabanı tablosu oluşturulması ve güncelleme
- Bulanık verilerin veritabanında ifade edilmesi
- Bulanık tablolar üzerinde ilişkisel cebir (Bulanık seçme, Bulanık sıralama, Bulanık kesişim ve Bulanık birleşim) işlemleri

Uygulamanın genel yapısı Şekil 5.1 de gösterilmiştir.



Şekil 5.1. Bulanık SQL uygulamasının genel yapısı

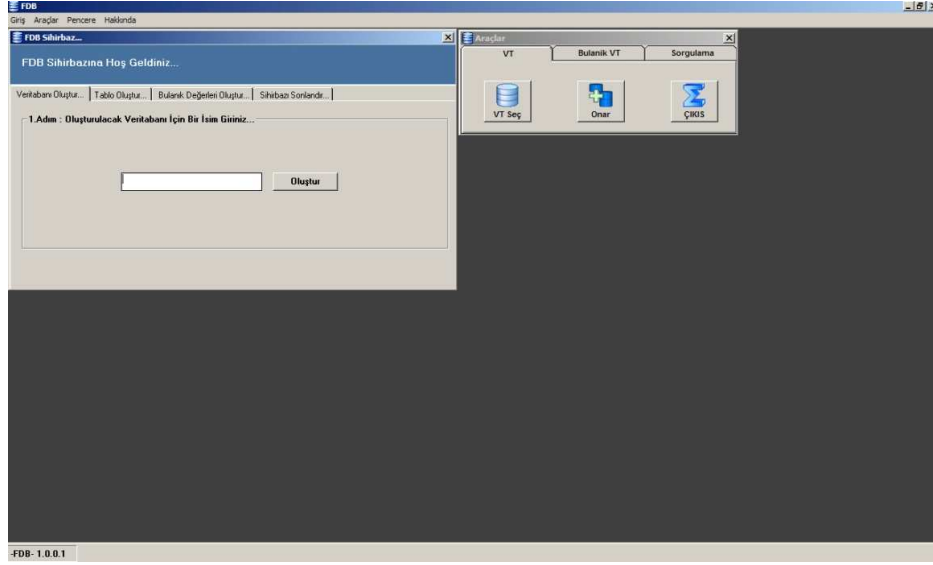
Bulanık SQL uygulaması klasik, bulanık sorgulama ve tablolar üzerinde ilişkisel cebir işlemlerini kullanmak amacıyla aşağıdaki alt programlardan oluşturulmuştur:

- Veritabanı oluşturma sihirbazı
- Araçlar
- Bulanık mantık işlemleri
- Bulanık sorgulama
- Klasik sorgulama
- Bulanık ve klasik sorgulama karşılaştırılması

Yukarıda gösterilen tüm alt programlara program ana sayfasından ilgili düğmeler kullanılarak ulaşılabilir.

5.1. Ana Ekran

Uygulamanın ana ekranının genel görünüşü şekil 5.2’ de gösterildiği gibidir. Uygulama ana ekranına kullanıcı adı ve şifre girilerek ulaşılır. Güvenlik için oluşturulan kullanıcı giriş ekranı kullanıcı adı “admin” şifre ise “12345” olarak oluşturulmuştur.

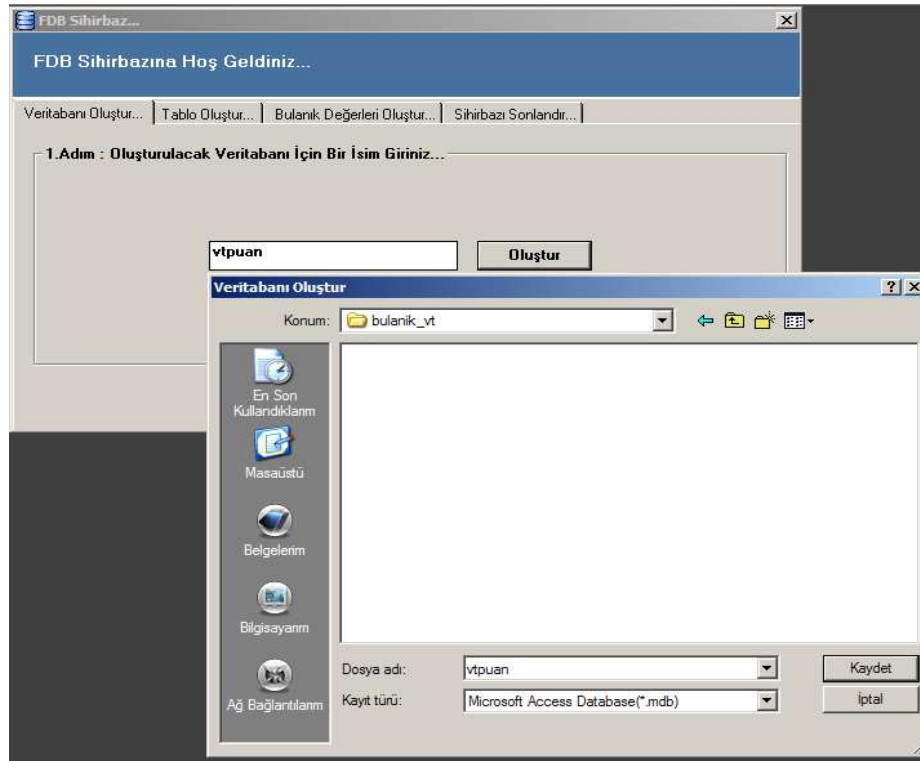


Şekil 5.2. Uygulamanın genel görünümü(Ana ekran)

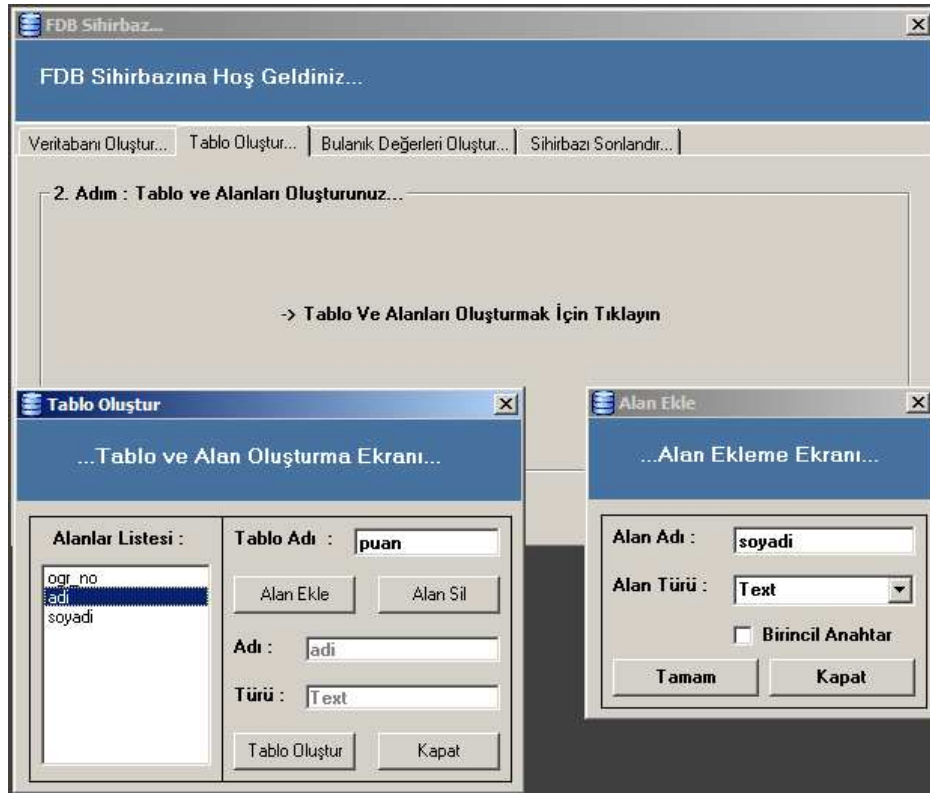
Veritabanı oluşturma sihirbazı ve araçlar alt programı uygulamanın ana ekranını oluşturan alt programlardır.

5.2. Veritabanı Oluşturma Sihirbazı

Veritabanı oluşturma sihirbazı ekranı uygulamamız için belirtilen isimde bir veritabanı oluşturur. “Tablo oluştur” sekmesine geçerek ilgili tablo oluşturularak alanlar eklenir ve bulanık değerler oluştur sekmesi ile küçükten büyüğe bulanık değerler girilir. Veritabanı oluşturma sihirbazı aracılığıyla veritabanı oluşturulması, tablo ve alanların eklenmesi, bulanık değerlerin girilmesi ekranlarının genel görünüşü aşağıdaki şekillerde gösterilmiştir.



Şekil 5.3. Veritabanı Oluşturma

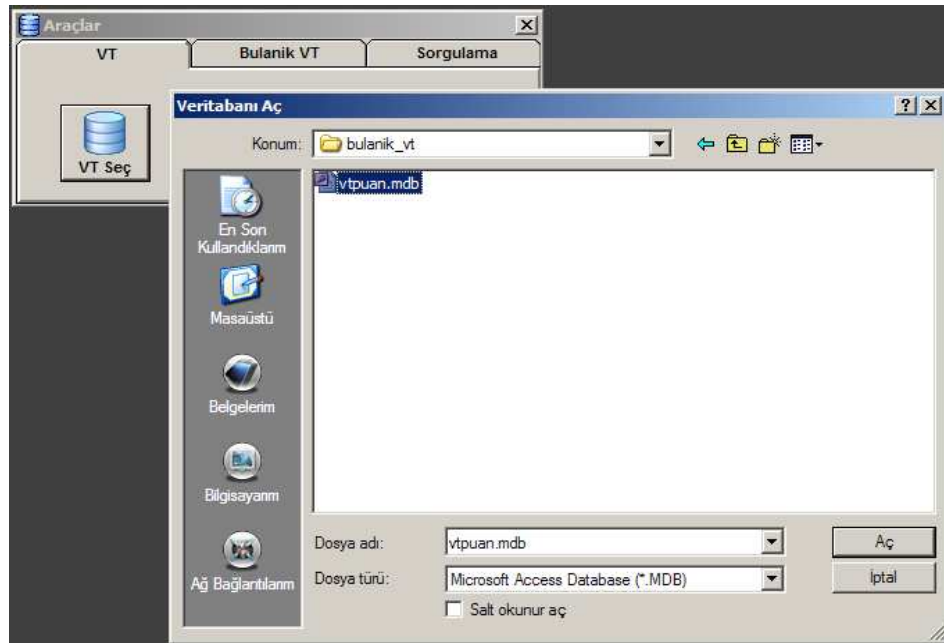


Şekil 5.4. Tablo ve alanların oluşturulması



Şekil 5.5. Bulanık değerler giriş ekranı

Sihirbazın son aşamasında bulanık değerler oluşturulduktan sonra araçlar penceresi kullanılarak oluşturulan veritabanı seçilir ve veriler üzerinde ekleme, silme, klasik ve bulanık sorgulama işlemleri yapılabilir.



Şekil 5.6. Araçlar kutusu veritabanı seçimi

Oluşturulan veritabanı seçildikten sonra klasik ve bulanık mantık işlemleri yapılabilir.

5.3. Bulanık Mantık İşlemler Programı

Uygulamanın bulanık mantık işlemler penceresi ile bulanık değerler tablosu okunur ve listelenir. Eğer değişiklik yapılmak istenirse yeni kayıt ekleme ya da silme gibi “Ekle” ve “Sil” düğmeleri kullanılır.

BD ADI	X1	X2	X3	X4
kötü	0	0	45	55
orta	50	60	68	75
iyi	70	75	85	90
pekiyi	85	90	100	100

ocr_no	adi	soyadı	kriter	fpuani	fderece
50801044	AHMET	VARAN	60	orta	1
50801047	MERT	DAĞLI	99	pekiyi	0
50801035	ERDEM	EREN	54	kötü	0,1
50801037	İSMAIL	YILDIZ	90	iyi	0
50801036	FATİH	HAKKI	88	iyi	0
50801034	ALİ	SÖNMEZ	73	orta	0,28
50801039	AHMET	KUÇUK	47	kötü	0,8
50801041	MURAT	BAŞAN	83	iyi	1

Şekil 5.7. Bulanık mantık işlemleri ekranı

Programın alt kısmında oluşturulan veritabanı tabloları listelenir ve listelenen tablolar seçilerek veriler ekrana yansıtılır. Bu bölüme istenirse yeni kayıt eklenir, değiştirilir ve silinebilir.

Seçilen tablonun kriter alanı (örnek tablomuzda kriter=puan olarak verilmektedir) seçili duruma getirilip programın “Hesaplama” bölümünden bulanık derece ve puanı (fderece ve fpuan) hesaplanır. “Tabloya ekle” düğmesiyle ilgili alanlara (fpuan ve fderece) veriler eklenir.

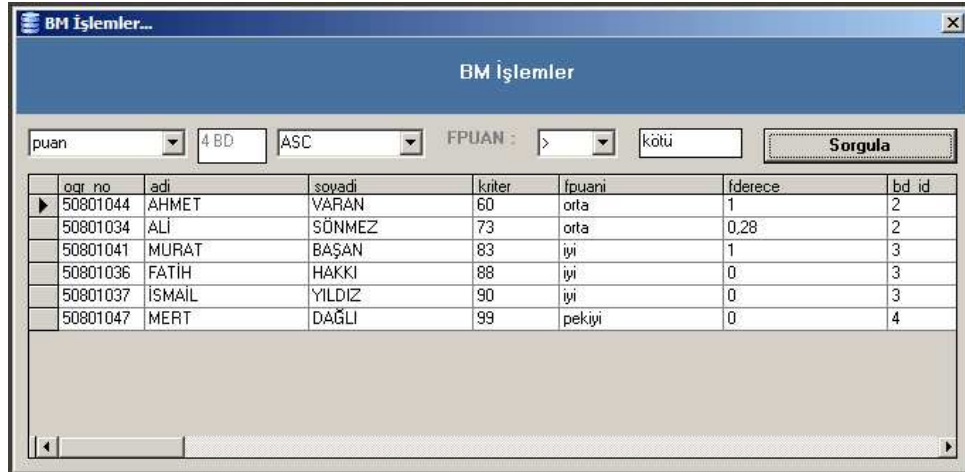
İşlemler düğmesi ise bu veriler üzerinde bulanık sorgulamalar yapmak için kullanılır.

5.4. Bulanık Sorgulama

Uygulamanın bir diğer alt programı bulanık sorgulamadır. Bu programın kullanılması için veritabanının, verileri ile birlikte tam hazır durumda olması gerekmektedir. Programa araçlar menüsünden veya bulanık mantık işlemleri penceresindeki ilgili düğmesinden ulaşılabilir. Programda sorgulama yapılacak tablo seçilir, sıralama koşulu verilir ve bulanık değere göre sorgulama yapılır. Örnek vermek gerekirse;

*SELECT * FROM PUAN WHERE FPUAN > "KÖTÜ"*

Örnek sorguda notuna göre kötü olmayan öğrenciler listelenir ve sıralanır. Şekil 5.8 'da bulanık sorgulama penceresinin genel görünümü gösterilmiştir.

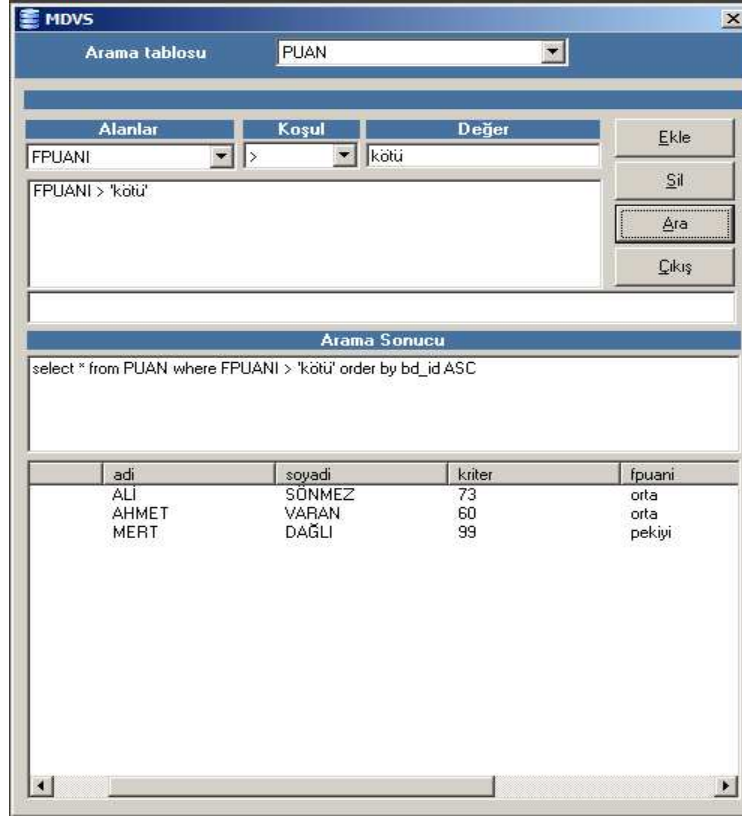


oqr no	adi	soyadi	kriter	fpuani	fderece	bd id
50801044	AHMET	VARAN	60	orta	1	2
50801034	ALI	SÖNMEZ	73	orta	0,28	2
50801041	MURAT	BAŞAN	83	iyi	1	3
50801036	FATİH	HAKKI	88	iyi	0	3
50801037	İSMAIL	YILDIZ	90	iyi	0	3
50801047	MERT	DAĞLI	99	pekiyi	0	4

Şekil 5.8. Bulanık Sorgulama Ekranı

5.5. Klasik ve Bulanık Sorgulamaların Karşılaştırılması

Klasik sorgulama programına araçlar menüsünden “Sorgu Oluştur” seçilerek yada klavye tuş takımından “Ctrl+Q” kullanılarak ulaşılabilir. Programın genel görünümü aşağıdaki şekilde verilmiştir.



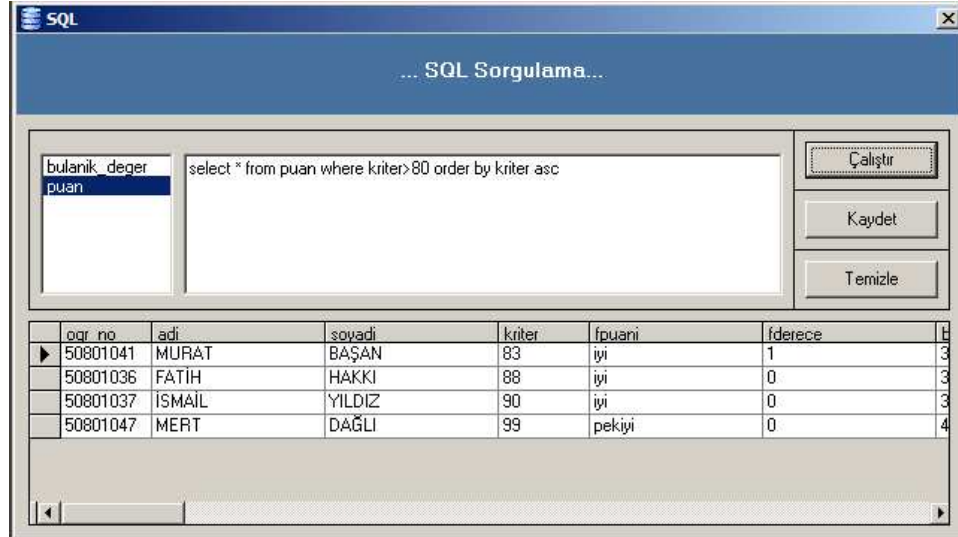
Şekil 5.9. Klasik Sorgulama Ekranı

Klasik sorgulama ekranında bulanık sorgulama ekranında olduğu gibi sorgulama yapılacak tablo seçilir, sorgulama koşulu hangi alanda yapılacaksa seçilir, “Ekle” düğmesine basılır ve “Ara” düğmesiyle sorgulama yapılır. Klasik sorgulama ekranında birden fazla koşul verilerek de sorgulama yapılabilir. Örnek olarak bulanık sorgulama programında kullandığımız sorguyu kullanalım.

*SELECT * FROM PUAN WHERE FPUAN > "KÖTÜ"*

Şekil 5.9’ da görüldüğü gibi klasik sorgulamada daha farklı bir sonuçla karşılaşırız. Bunun sebebi; ilgili sorguda “FPUAN > KÖTÜ ” alfabetik düzene göre “KÖTÜ” ‘den büyük (k harfinden büyük) kayıtlar listelenir.

Klasik sorgulamaya ek olarak SQL sorgulama ekranı geliştirilmiştir. Bu programın genel görünümü şekil 5.10’ de gösterilmiştir. İlgili programda SQL sorgusu manüel olarak yazılır.



Şekil 5.10. SQL Sorgulama ekranı

SİMGELER VE KISALTMALAR LİSTESİ

Simge	Açıklama
\cap	Kesişim
\cup	Birleşim
\subset	Kapsar
\subseteq	Kapsar veya eşit
\in	Elemanıdır
\notin	Elemanı değildir
μ	Üyelik fonksiyonu
\otimes	Min-max kompozisyonu
Ω	O andaki pozisyon değeri
\wedge	VE
\vee	VEYA
$\bar{\cdot}$	Bulanık DEĞİL
$\bar{\wedge}$	Bulanık VE
$\bar{\vee}$	Bulanık VEYA
[]	Kapalı aralık
{ }	Küme işareti
$>$	Büyük
\geq	Büyük eşit
$<$	Küçük
\leq	Küçük eşit

≠	Eşit değildir
U	Evrensel küme
VT	Veri Tabanı
VTYS	Veri Tabanı Yönetim Sistemi
SQL	Structured Query Language (Yapısal Sorgulama Dili)
SQLf	Bulanık yapısal sorgulama dili (SQL fuzzy)
FRDB	Fuzzy Relational Databases (Bulanık İlişkisel Veritabanları)
DDL	Data Definition Language (Veri Tanımlama Dili)
JDBC	Java Database Connectivity (Java Veritabanı bağlantı köprüsü)
UFO	Uncertainty and Fuzziness in an Object-oriented (Nesneye Yönelikte Belirsizlik ve bulanıklık)
BNYVT	Bulanık Nesneye Yönelik Veri Tabanı
OMRON	Oracle VT arayüzü için geliştirilmiş sistem
FQUERY	Fuzzy Query (Bulanık Sorgu)
AIS	Automatic Identification System (Otomatik Tanımlama Sistemi)
OMT	Object-modeling technique (Nesneye yönelik yöntem)
GNO	Genel Not Ortalaması
v.b.	Ve benzeri
v.d.	Ve diğerleri
yy	Yüzyıl
n.d.	No date (bilinmeyen tarih)
FL	Fuzzy Logic (Bulanık Mantık)

ÇİZELGELER LİSTESİ

	Sayfa
Çizelge 3.1. Uzman1’in değerlendirmesi-U1 Çizelgesi.....	17
Çizelge 3.2. Uzman 2’in değerlendirmesi-U2 Çizelgesi.....	17
Çizelge 3.3. U1 ve U2 çizelgelerinin birleşimi $U1 \cup U2$	18
Çizelge 3.4. U1 ve U2 ilişkilerinin kesişimi $U1 \cap U2$	19
Çizelge 3.5. Küçültülmüş U1 ilişkisi	19
Çizelge 3.6. Küçültülmüş U2 ilişkisi	19
Çizelge 3.7. U1K ve U2K ilişkilerinin Kartezyen çarpımı	20
Çizelge 3.8. Y matrisi.....	21
Çizelge 3.9. K matrisi.....	21
Çizelge 3.10. Y ve K ilişkilerinin kesişimi	21
Çizelge 3.11. Bulanık birleşim sonucu	22
Çizelge 3.12. R ilişkisi	23
Çizelge 3.13. r bulanık ilişkisi.....	25
Çizelge 3.14. s bulanık ilişkisi	25
Çizelge 3.15. Yaş özelliği üzere yakınlık ilişkisi.....	26
Çizelge 3.16. r ve s ilişkilerinin birleşimi $r \cup s$	26
Çizelge 3.17. r ve s ilişkilerinin farkı.....	26
Çizelge 3.18. S ilişkisi.....	27
Çizelge 3.18. S ilişkisinin izdüşümü.....	27

Çizelge 3.19. r ilişkisi.....	28
Çizelge 3.20. Seçme işlemi sonucu.....	28
Çizelge 3.21. r ilişkisi.....	28
Çizelge 3.22. s ilişkisi	29
Çizelge 3.23. r ve s ilişkilerini doğal bitleştirilmesi sonucu	29
Çizelge 4.1. Bulanık ve belirsiz veriler içeren KİŞİ ilişkisi.....	38
Çizelge 4.2. Benzerlik matrisi örneği.....	40
Çizelge 4.3. Özellik ve üyelik derecesi.....	41
Çizelge 4.4. Öğrenci çizelgesi.....	42

ŞEKİLLER LİSTESİ

	Sayfa
Şekil 2.1. Bulanık Veritabanının temel bileşenleri	4
Şekil 2.2. X ve A kümelerinin ifadesi	6
Şekil 2.3. A kümesi ve x değerlerinin bu kümeye ait olup olmadığını gösteren $\chi_A(x)$ ilişki fonksiyonu	6
Şekil 2.4.a. Kesin değerli A kümesi	8
Şekil 2.4.b. Bulanık değerli A kümesi	8
Şekil 2.5. Kesin ve bulanık A kümeleri	8
Şekil 2.6. A bulanık kümesinin desteği $[a_1(0), a_2(0)]$	9
Şekil 2.7. Bulanık kümenin üçgen ifadesi	10
Şekil 2.8. Bulanık kümenin yamuk ifadesi	11
Şekil 2.9. Kesin kümelerde işlemlerin grafiksel ifadesi: a) kesişim, b) birleşim c) tamamlama	11
Şekil 2.10. Bulanık tamamlamanın grafiksel ifadesi	12
Şekil 2.11. Bulanık birleşim işleminin grafiksel ifadesi	12
Şekil 2.12. Bulanık kesişim işleminin grafiksel ifadesi	13
Şekil 2.13. Yığılma işleminin grafiksel ifadesi	13
Şekil 2.14. Genişlenme işleminin grafiksel ifadesi	14

Şekil 4.1. PFSQL sisteminin mimarisi	31
Şekil 4.2. OMRON Bulanık Veritabanı mimarisi	34
Şekil 4.3. Bulanık veritabanı modellerinin sınıflandırılması	35
Şekil 4.4. Kesin, kesin olmayan ve muğlak verilerin grafik tasviri	37
Şekil 4.5. Yaklaşık değerin üçgen ifadesi	39
Şekil 4.6. KÜÇÜK sözel teriminin mümkünlük dağılımının yamuk ifadesi	39
Şekil 4.7. Genç, orta yaşlı ve yaşlı bulanık kümelerinin tanımlanması	42
Şekil 4.8. a) YAŞ ve b) GNO sözel değerlerinin grafik ifadesi	42
Şekil 4.9. Bulanık veritabanının metaveri yapısı	44
Şekil 4.10 Bulanık ER diyagramı.....	48
Şekil 4.11. Bulanık toplama	48
Şekil 5.1. Bulanık SQL uygulamasının genel yapısı.....	50
Şekil 5.2. Uygulamanın genel görünümü(Ana ekran)	51
Şekil 5.3. Veritabanı Oluşturma.....	52
Şekil 5.4. Tablo ve alanların oluşturulması.....	52
Şekil 5.5. Bulanık değerler giriş ekranı.....	53
Şeki 5.6 Araçlar kutusu veritabanı seçimi.....	53
Şekil 5.7. Bulanık mantık işlemleri ekranı.....	54

Şekil 5.8. Bulanık Sorgulama Ekranı	55
Şekil 5.9. Klasik Sorgulama Ekranı	56
Şekil 5.10. SQL Sorgulama ekranı.....	57

KAYNAKLAR

- Awadhesh K.S., Goswami, A., Gupta, D.K., 2009, July 1-3, *FTS-SQL: A Query Language for Fuzzy Multidatabases*, Proceedings of the World Congress on Engineering 2009 Vol I, WCE 2009, London, U.K.
- Bosc, P. ve Pivert, O., 1995, SQLf: a relational database language for fuzzy querying, *IEEE Transactions on Fuzzy Systems*, 3(1): 1-17.
- Bosc, P. ve Pivert, O., Extending SQL retrieval features for the handling of flexible queries. In: Dubois, D., Prade, H., Yager, R.R. (Eds.), *Fuzzy Information Engineering: A Guided Tour of Applications*, Wiley Computer Publishing. pp. 233-251.
- Chen, G. Q., 1999, *Fuzzy Logic in Data Modeling, Semantics, Constraints, and Database Design*, Kluwer Academic Publisher.
- Fuller R., *Fuzzy relations*, (b.t.), Retrieved May 12, 2009 from <http://users.abo.fi/rfuller/nfs2.pdf>
- Fuller R., *Fuzzy Sets and Fuzzy Logic*, (b.t.) May 12,2009 from <http://users.abo.fi/rfuller/nfs1.pdf>
- Galindo J., 2006, *Handbook of Research on Fuzzy Information Processing in Databases*, IGI Publishing, Hershey, PA, USA.
- Gyseghem V., Caluwe N. D., Vandenberghe R., 1993, *Fuzzy Systems*, UFO: uncertainty and fuzziness in an object-oriented model, Second IEEE International Conference on Volume , Issue , p: 773 – 778.
- Hung T. N ve Elbert A. W., A First Course in Fuzzy Logic. *Boca Raton, FL: CRC Press*, 1997. 266 pp.
- Larsen L. H., 2005, *Introduction. Fundamentals of fuzzy sets and fuzzy logic*. Course in Fuzzy Logic Lecture 1:,Aalborg University Esbjerg, AAUE Computer Science
- Liangzhong, D. W. Ve Pei Y. Z., October 2007, *Application of Fuzzy Query Based on Relation Database*, ISKE-2007 Proceedings.

- Liu, W. Y., 1993, Extending the relational model to deal with fuzzy values, *Fuzzy Sets and Systems*, 60 (2): 207'-212.
- Ma, Z., 2005, *Fuzzy Database Modeling with XML*, Universite de Sherbrooke, Canada, Advances in Database Systems Volume 29.
- Ma, Z., 2005, A conceptual design methodology for fuzzy relational databases, *Journal of Database Management*, Idea Group Publishing, Hershey, PA, USA.
- Ma, Z., 2004, *Advances in Fuzzy Object-Oriented Databases: Modeling and Applications [Illustrated] (Hardcover)*, Idea Group Publishing; illustrated edition.
- Ma Z., Yan L., 2008. A Literature Overview of Fuzzy Database Models, *Journal of information science and engineering* 24: 189-202
- Nathan, J., (b.t) Fuzzy Lookups and Groupings Provide Powerful Data Cleansing Capabilities, *SQL Server 2005*, Microsoft Corporation, pp. 1-6.
- Petry, E.F., 1996. *Fuzzy Databases: Principles and Applications*. Kluwer Academic Publishers, Boston.
- Punam B., Harmeet K., Malhotra A., 2002, *Fuzzy Dimension To Databases*, Proc. 37th Annual Convention of Computer Society of India.
- Surajit C., Ganjam K., Venky G., Narasayya V., Vassilakis T, September 2005, Microsoft SQL Server 9.0 Technical Articles, Fuzzy Lookup and Fuzzy Grouping in SQL Server, Integration Services 2005, Microsoft Corporation, retrieved October 21, 2009 from [http://msdn.microsoft.com/en-us/library/ms345128\(SQL.90\).aspx](http://msdn.microsoft.com/en-us/library/ms345128(SQL.90).aspx)
- Takači A., Skrbic S, 2008, Priority, Weight and Threshold in Fuzzy SQL Systems, *Acta Polytechnica Hungarica* Vol. 5, No. 1, 2008.
- Urrutia A. ve Pavesi L., October 2004, *Extending the capabilities of database queries using fuzzy logic* , Catholic University of Maule (CHILE), Retrieved September, 26, 2009, from http://www.collector.org/archives/2004_October/06.pdf
- Yang Q., Zhang W., Liu C, Wu J, Hiroshi N, Rische N.D., November/December 2001, Efficient Processing of Nested Fuzzy SQL Queries in a Fuzzy Database, *IEEE Transactions on Knowledge and Data Engineering*, Volume 13, NO. 6, 884-901

Zadeh, L. A., 1965, Fuzzy sets, *Information and Control*, 8 (3): 338-353.

Zvieli, A. ve Chen, P. P., 1986, Entity-relationship modeling and fuzzy databases, *Proceedings of the 1986 IEEE International Conference on Data Engineering*, p.320-327.

ÖZGEÇMİŞ

KİŞİSEL BİLGİLER:

Adı Soyadı: Vügar SALAHLI

Doğum Yeri: Azerbaycan, Bakü

Doğum Tarihi: 23.01.1984

EĞİTİM DURUMU:

Lisans Öğrenimi: Çanakkale Onsekiz Mart Üniversitesi, Mühendislik Mimarlık Fakültesi

Yüksek Lisans Öğrenimi: Çanakkale Onsekiz Mart Üniversitesi, Fen Bilimleri Enstitüsü, Bilgisayar Mühendisliği Anabilim Dalı

Bildiği Yabancı Diller: İngilizce, Rusça

İŞ DENEYİMİ:

Figen Yazılım Evi, Web Yazılımı Uzmanı, Programcı, 2002-2003

Teknolojix Bilgisayar, Teknik Servis Şefi, Donanımcı, 2004-2005

Hızlı Sistem Bilgisayar, Teknik Servis Şefi, 2005-2006

Universal Bilgi Teknolojileri, Bölge KBS Sorumlusu, 2006-2009

İLETİŞİM

E-posta adresi: vugarsalahli@yahoo.com, vugarsalahli@gmail.com

Cep Tel: (0544) 367 99 11

Ev Tel: (0286) 218 14 61